Information Fusion 80 (2022) 23–43

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Full length article

Hierarchical fusion and divergent activation based weakly supervised learning for object detection from remote sensing images

Zhi-Ze Wu [a], Jian Xu [b], Yan Wang [c], Fei Sun [d], Ming Tan [d], Thomas Weise [a,*]

[a] Institute of Applied Optimization, School of Artificial Intelligence and Big Data, Hefei University, Jinxiu Dadao 99, Hefei, Anhui, 230601, China
[b] School of Computer Science and Technology, University of Science and Technology of China, Hefei, Anhui, 230027, China
[c] School of Art, Anhui Jianzhu University, Jinzhai Road 856, Hefei, Anhui, 230022, China
[d] School of Artificial Intelligence and Big Data, Hefei University, Jinxiu Dadao 99, Hefei, Anhui, 230601, China

ARTICLE INFO

Keywords:
Hierarchical fusion
Object detection from remote sensing images
Weakly supervised learning
Class activation map
Divergent activation

ABSTRACT

Object detection and location from remote sensing (RS) images is challenging, computationally expensive, and labor intense. Benefiting from research on convolutional neural networks (CNNs), the performance in this field has improved in the recent years. However, object detection methods based on CNNs require a large number of images with annotation information for training. For object location, these annotations must contain bounding boxes. Furthermore, objects in RS images are usually small and densely co-located, leading to a high cost of manual annotation. We tackle the problem of weakly supervised object detection under such conditions, aiming to learn detectors with only image-level annotations, i.e., without bounding box annotations. Based on the fact that the feature maps of a CNN are localizable, we hierarchically fuse the location information from the shallow feature map with the class activation map to obtain accurate object locations. In order to mitigate the loss of small or densely distributed objects, we introduce a divergent activation module and a similarity module into the network. The divergent activation module is used to improve the response strength of the low-response areas in the shallow feature map. Densely distributed objects in RS images, such as aircraft in an airport, often exhibit a certain similarity. The similarity module is used to improve the feature distribution of the shallow feature map and to suppress background noise. Comprehensive experiments on a public dataset and a self-assembled dataset (which we made publicly available) show the superior performance of our method compared to state-of-the-art object detectors.

1. Introduction

With the continuous development of modern remote sensing (RS) technology, many RS images with high spatial resolution are regularly produced and provide data for various fields of research [1–4]. Object recognition from images means to automatically find the object(s) of interest and to return their category and location information. Due to the continuous improvement of the spatial resolution of RS images, the information contained in single images is growing. The currently existing automated image interpretation methods are unable to meet the needs of many real-world applications in the RS community [5,6]. Therefore, the question of how to accurately extract the position and class of objects from RS images has come into the focus of research.

During the past few years, deep learning has received growing attention, as it can automatically discover problem-specific features for object detection. Convolutional neural networks (CNNs) [7] are the most widely used deep-learning method. With the development of CNNs, the accuracy of object detection from RS images has been boosted significantly [8,9]. In [10], for instance, Ding et al. adopted a multi-scale representation, a dense convoluted network, and various combinations of improvement schemes to enhance the feature learning ability of the VGG-16 network [11], an often-used CNN. Based on the backbone of ResNet-50 [12], Zhen et al. [5] proposed a hyper-scale object detection framework named HyNet, which has several multi-scale structures within the convolutional layer to alleviate the extreme scale variation problem by learning hyper-scale feature representations.

High-quality and large-scale datasets are important for the training of CNN-based object detection methods. Most often, fully supervised training is performed, which requires a large amount of training data annotated with a bounding box around each object instance. However,

* Corresponding author.
E-mail addresses: wuzhize@mail.ustc.edu.cn (Z.-Z. Wu), xxxujian@mail.ustc.edu.cn (J. Xu), YanWang0417@ahjzu.edu.cn (Y. Wang), sunfei@hfuu.edu.cn (F. Sun), tanming@hfuu.edu.cn (M. Tan), tweise@gmx.de (T. Weise).
URL: https://iao.hfuu.edu.cn (Z.-Z. Wu).

https://doi.org/10.1016/j.inffus.2021.10.010
Received 13 January 2021; Received in revised form 13 October 2021; Accepted 31 October 2021
Available online 13 November 2021

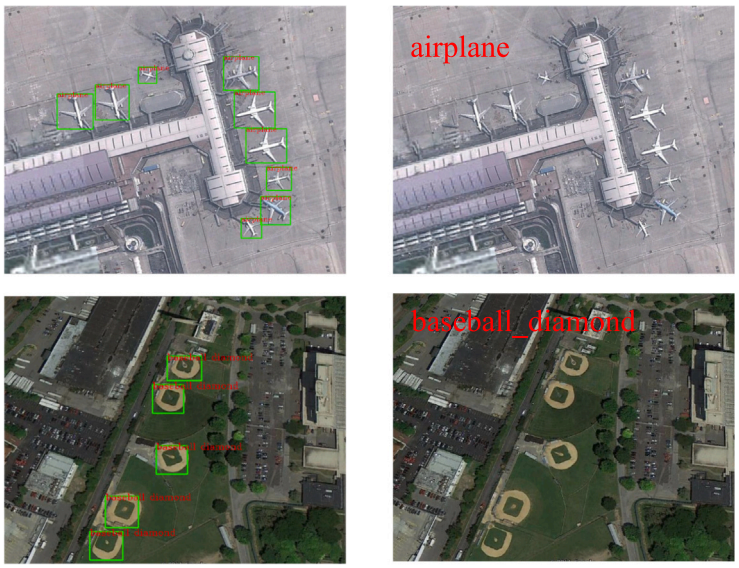


Fig. 1. Instance-level annotation and image-level annotation.

collecting such precise instance-level annotations is labor-intensive and time-consuming.

Annotations in the form of image-level labels, as shown on the right-hand side of Fig. 1, are much easier to obtain compared to instance-level annotations. As remedy for the drawbacks of fully supervised training [5,9,10,13], several studies therefore focus on weakly supervised object detection (WSOD), where the training set only needs binary labels indicating whether an image contains the target object or not.

Every convolutional unit in the CNN is essentially a detector that can locate the target object in the image [14]. For example, if the object appears in the upper left corner of the image, the upper left corner of the feature map after the convolutional layer will produce a greater response. If the target object is in the lower right corner, the region in the lower right corner of the feature map will have a larger response. Zhou et al. [15] showed that feature maps extracted by a CNN are indeed localizable representations of the image. Based on this fact, several class activation map (CAM) based WSOD methods have been developed. In order to solve the problem that the target area still cannot be completely and accurately located, researchers proposed positioning optimization methods based on highlighted area erasure [16,17], on seed highlighted area editing [18,19], and on the expansion of the highlighted response area [20,21].

Recently, efforts have been made to apply CAM-based WSOD to RS images [22–24]. For instance, Qiao et al. [22] employ a weakly supervised localization method for detecting red-attacked trees in aerial images. Weakly supervised training of a CNN is also used in [23], where Wu et al. propose a new aircraft detection model for RS images called AlexNet-WSL. In [25], Li et al. present a multi-scale scene-sliding-voting strategy to calculate the CAM of RS images. They use the mutual information between scene pairs to train deep networks under scene-level supervision for multi-class geospatial object detection.

CAM based weakly supervised learning methods are especially effective for images of natural scenes, where the objects are relatively large and their number is small. In RS images, however, the objects cover small image areas, there are both small and large objects, and the distribution of the small object is often dense. Due to the limitation of the activation map to local characteristics, it is difficult to improve the positioning and detection performance for such densely co-located, small objects.

A typical example application where this problem occurs is aircraft detection. Fig. 2 shows an image containing several aircraft in an airport. When the CNN extracts features, the response regions contain the specific category characteristics to classify the objects. Multiple aircraft in one image might emphasize the class response, but if they are close to each other, the response region will encompass all of them. As a result, it is difficult to accurately locate the aircraft by only using the CAM from RS images. This is problematic, because we often want to detect aircraft that have landed, are located in airports, and hence, are densely distributed. Similarly, ships are often detected when being docked in ports and oil tanks also often appear in groups of many. Being able to detect the objects in such situations can decide about the feasibility of an RS application. This phenomenon of densely-distributed objects rarely occurs in natural scenes and, if it does, is often not as important as in the RS field.

Also, in natural scenes, the object relations in an image are mostly semantic [26–31], such as “a table and a sofa are in a room”. In RS images, however, such semantic information is difficult to obtain. It may not be possible to include both complete ports and ships in the section of a RS image that is processed by the object detector at a high resolution. Very large-scale RS image, which often cover huge areas of more than 10 km × 10 km, will be processed in slices with, for example, a size of 1000 × 1000 pixels. Under a low spatial resolution, these could indeed include large and small targets, like ports and ships, at the same time. But then it becomes difficult to perform object identification, especially of the smaller objects. Under high spatial resolutions, such as 3 m × 3 m per pixel, however, the ports basically become background areas in a 1000 × 1000 pixel RS image.

In this paper, we tackle the problem of WSOD from RS images with high spatial resolution. We aim to learn detectors with only image-level annotations, i.e., without object location information during the training stage. We experimentally confirm that the shallow feature maps of the network contain accurate object locations, but, at the same time, large background areas will be included in the response. This led us to the idea of hierarchically fusing the information of the shallow feature map with the CAM to obtain accurate object locations. We name this new method *Hierarchical Fusion Based Remote Sensing Object Location* (HF-RSOL).

We then further improve the HF-RSOL in order to mitigate the detection loss of small densely located objects. We introduce the novel

Fig. 2. Class activation map based object location in RS images.

combination of a divergent activation module and a similarity module, and propose a similarity-based divergent activation approach for object detection. We name this approach *Similarity-Divergent Activation Based Remote Sensing Object Detection* (SDA-RSOD). The divergent activation module is used to increase the response strength of the low-response areas in the shallow feature map. According to the characteristic high object density and the similarity between objects in RS images, the similarity module is used to improve the feature distribution of the shallow feature map and also to suppress background noise.

Comprehensive experiments on a public dataset and a self-assembled dataset (which we have made publicly available [23,32]) show the superior performance of our method compared to several state-of-the-art object detectors. All source code used in our experiments as well as the results are provided in the immutable online repository (https://zenodo.org/record/4420286). The main contributions of this paper are:

1. With our hierarchical feature map based object location method HF-RSOL, we propose an information fusion approach for object detection from RS images. HF-RSOL therefore combines the location information from the shallow feature map and the class activation map of a CNN to obtain accurate object locations.
2. We then extend HF-RSOL by proposing the similarity-divergent activation based object detection method SDA-RSOD. By integrating a similarity module and a divergent activation module, SDA-RSOD can effectively mitigate detection misses of small and densely distributed objects.
3. We evaluate both methods with comprehensive experiments and find that they are very effective for object detection from RS images. We also analyze the hyperparameter settings of the similarity module and the divergent activation module.

The remainder of this paper is organized as follows. Section 2 provides an overview of the related work on WSOD methods. In Section 3, we introduce our novel object location algorithm, and in Section 4, we present the object detection approach. In Section 5, we evaluate the algorithm performance. The conclusions and discussion are given in Section 6.

2. Related work

Object detection from RS images has many important aspects [33–35]. How to distinguish object instances from complex backgrounds is the main challenge for WSOD in RS. Two main branches of approaches for tackling WSOD have emerged, namely multiple-instance learning (MIL) and class activation map (CAM) based methods. We will discuss these two groups in the following text.

2.1. Methods based on multiple-instance learning

MIL treats each training image as a bag and iteratively selects highly-scoring instances from each bag when training the object detectors [36]. With the development of deep learning and especially CNNs, deep MIL networks have been studied due to their better performance.

Bilen et al. [37] propose a two-stream CNN based weakly supervised deep detection network (WSDDN), which selects positive samples by multiplying the detection and classification scores. In [38], Kantorov et al. introduce two kinds of context-aware guidance, namely additive and contrastive models, into the WSDDN for improving object localization. Tang et al. [39] propose an online instance classifier refinement (OICR) algorithm to integrate MIL and the instance classifier refinement procedure into a single deep network. They train this network end-to-end with only image-level supervision. By combining WSDDN [37] and OICR [39], Zhang et al. [40] designed a weakly-supervised to fully-supervised framework for object detection, where a weakly-supervised detector is implemented using MIL.

Wan et al. [41] introduce a continuation optimization method into MIL to create their continuation multiple instance learning (C-MIL) method with the intention to alleviate the non-convexity problem in a systematic way. In order to model the uncertainty in the location of the objects, Arun et al. [42] employ a dissimilarity coefficient based probabilistic learning objective, which minimizes the difference between an annotation agnostic prediction distribution and an annotation aware conditional distribution. Yang et al. [43] design a single network with both multiple instance learning and bounding-box regression branches that share the same backbone. This avoids the two-phase learning procedure, i.e., training the multiple instance learning detector followed by the fully supervised learning detector with bounding-box regression. Shen et al. [44] join weakly supervised object detection and segmentation tasks with a multi-task learning scheme, which uses their respective failure patterns to complement each other.

Although promising results have been reported for the aforementioned WSL-based object detection in natural scenes, these methods should not directly be applied to RS images. Here, it is more challenging to detect object instances because they often occur within close proximity of each other and only occupy a small proportion of large images

with complex backgrounds. Furthermore, different scales and rotations lead to large variations in appearance, which add to the complexity of the problem.

The latest works on deep MIL based object detection from RS images are [45–47]. In [45], Feng et al. develop an end-to-end progressive contextual instance refinement model (PCIR) to perform WSOD. This model can divert the focus of the detection network from locally distinct parts to the object and then further to other potential instances by leveraging both local and global context information. The work [46] proposes a dynamic curriculum learning strategy to progressively learn the object detectors by feeding training images with an increasing difficulty that matches the current detection ability. With this, the detectors are gradually improved more effectively. In [47], Feng et al. design a unique end-to-end WSOD network for learning to learn complementary and discriminative visual patterns in RS images. This network consists of a global context-aware enhancement (GCAE) module and a dual-local context residual (DLCR) module. The GCAE can activate the features of the whole object by capturing the global visual scene context. And the DLCR model uses an effective adaptive-weighted refinement loss function to reduce the ambiguities in the label propagating process.

MIL-based approaches are in principle suitable for WSOD tasks. However, the additional cost of step-by-step training is much higher compared to an end-to-end structure. Various optimization strategies are used to adjust for this, but the adaptability of the algorithm itself is very limited. For different types of data, it may be necessary to use several optimization strategies to adjust parts of the algorithm, such as how to initialize and how to extract features to represent an instance bag. It is difficult to adapt MIL methods to the complexity and diversity of image data in the real world, which makes it also difficult to transfer trained networks to new types of scenes. We will compare the performance of our approach to WSDDN, OICR, and PCIR.

2.2. Methods based on class activation maps

Another approach for tackling WSOD is to formulate it as a localizable feature map learning problem by weighting the output of the feature map of a CNN according to the input image category and then iteratively selecting high-response areas when learning the object detectors [19–21,23,25].

The first attempt to utilize CAMs for object detection was proposed in [15]. This method prescribes pre-training a classification CNN. It adds a global average pooling (GAP) layer and a fully connected layer to the CNN. Finally, it combines the final convolution layer features with the weights of the fully connected layer to generate an object location map, that is, a class activation map. This way, end-to-end CNNs are trained. By fine-tuning their weights, networks can be transferred to different scenarios, thereby improving the applicability of the algorithm.

In [14], Ren et al. demonstrated that every convolutional unit in the CNN is essentially a detector that can locate a target object in the image. Objects located in the upper-left corner of an image will lead to a greater response in the upper-left corner of the feature map after the convolutional layer. Objects in the lower-right corner instead causes stronger responses in the lower-right region of the feature map, and so on. The works of Qiao et al. [22], Li et al. [25], and Wu et al. [23] already mentioned in the introduction make use of this concept.

Aiming at the problem that objects still cannot be located accurately, Wei et al. [16] propose an adversarial erasing approach for localizing and expanding object regions progressively. It drives the classification network to sequentially discover new and complement object regions by erasing the currently mined regions in an adversarial manner. Similar to [16], Zhang et al. [17] propose a novel Adversarial Complementary Learning (ACoL) approach to efficiently mine different discriminative regions by two adversary classifiers in a weakly supervised manner, which discover integral target regions of objects for localization. We will compare the performance of our approaches to this seminal work from 2018.

In [24], Li et al. propose a weakly supervised deep learning-based cloud detection method using block-level labels indicating only the presence or the absence of clouds in one RS image block. In order to improve the quality (e.g., spatial resolution) of the CAM, a local pooling pruning strategy is applied to prune the local pooling layers in the training phase.

Kim et al. [18] propose a two-phase learning method for weakly supervised object localization. During the first phase, a conventional FCN is trained for image-level classification. At this time, the pixels belonging to the most important parts in an image are revealed in a heat map. During the second phase, the activations of these parts are suppressed by inference conditional feedback. Then the second learning round is performed to find the area of the next most important parts. By combining the activations of both phases, the entire portion of the target object can be captured.

Yun et al. [19] propose a stage-wise approach to learn high-quality self-produced guidance masks, which distinguish the foreground and background of a given image. By integrating self-produced supervision into the weak object localization, this method can help the classification network to discover pixel correlations to improve the localization performance.

Xue et al. [48] design the hierarchical divergent activation (HDA) and discrepant divergent activation (DDA) ideas. The goal here is to learn complementary and discriminative visual patterns for image classification and to perform weakly supervised object localization from the perspective of discrepancy. These DANets – deep networks extended with HDA and DDA – diverge and fuse discrepant yet discriminative features for image classification and object localization in an end-to-end manner. We will compare the performance of our new methods with this recent state-of-the-art approach, which was published in 2019.

In [49], Rey-Area et al. propose class-inherent transformation generators for improving the generalization capacity of image classification models, especially appropriate for problems in which the involved classes share many visual features. Further methods to expand the response region and then to locate the complete areas of the objects are proposed in [20,21,50].

The above-mentioned CAM-based weakly supervised learning methods have difficulties to achieve good positioning and detection performance for small and densely co-located objects, i.e., exactly in the setting occurring in the RS scenario. One notable exception is the AlexNet-WSL [23] published in 2020, which uses the AlexNet CNN [51] as backbone network, but replaces the last two fully connected layers with a GAP and two convolutional layers. It generates heat maps from the CAM via reverse weighting for locating the target objects. This way, this weakly-supervised approach achieved a performance on WSADD equivalent to what YOLOv3 [52] and Faster R-CNN [14] trained in a fully supervised manner deliver. We will use this approach as benchmark for comparison.

Different from AlexNet-WSL, we perform hierarchical response fusion and introduce a divergent activation (DA) module on the basis of DANet [48]. This can improve the response intensity of unresponsive and low-response areas. Objects that are small in size, large in number, and occur closely co-located in an image, e.g., aircraft in an airport, are often similar. We use this fact to suppress the background noise in the third layer of the network by applying the dual attention mechanism from [53,54] as similarity module. By constructing a similarity matrix of the response feature maps, we can improve the response intensity of the similar areas and thus the feature distribution in the shallow feature map. By combining a similarity measure and the divergent activation, the problem of detection misses of such object in RS images is effectively mitigated.

3. Hierarchical fusion based object location from remote sensing images

Aiming to improve the accuracy of object location for RS images, we fuse the shallow feature map and the CAM. In Section 3.1, we present the framework of this new *Hierarchical Fusion Based Remote Sensing Object Location* (HF-RSOL) method. The architecture and the training mechanism of the network are given in Section 3.2. In Section 3.3, we introduce how to generate the object location map using the HF-RSOL.

3.1. Information fusion framework of the HF-RSOL

There are two information sources that drive the HF-RSOL. The first one is the final convolutional layer of the CNN, which often combines multiple response regions in order to improve the classification performance. These highlighted response regions are related to the locations of the objects. For example, multiple aircraft could create separate responses, but after the final convolutional layer, these regions would all be merged into one, which may lead to incorrect positioning. Second, the learned features of the shallow layer contain the accurate location of the target object, but at the same time there will be a large number of background regions in the response. We fuse the location information of the shallow feature map with the class activation map in our HF-RSOL to obtain accurate object locations.

We illustrate the information fusion framework of HF-RSOL in Fig. 3. The HF-RSOL utilizes a basic CNN to extract the features of the input image and convert the network output to the feature maps with a size of $C \times H \times H$ using 1×1 convolutions, where C is the number of object categories. Based on the preset confidence threshold, we obtain the class activation map (CAM) from the deep feature maps. We use the feature map output of the shallow layer as another class activation map, which contains the accurate object location. We refer to these maps as deep class activation map (DCAM) and shallow class activation map (SCAM), respectively. In the stage of generating the final object location map, we first perform threshold segmentation on the DCAM image and the SCAM image into corresponding binary images.

The dot product operation is applied to the binarized images to obtain the fused location map. For more than 40 years, the dot product is used to calculate matrix similarities and to perform simple image fusion [55]. Recently, Jin et al. [56] propose to compute the saliency image by computing the matrix dot product of the original image and their saliency map. Do et al. [57] apply the dot product to select a representative subset of local convolutional features and eliminate redundant features. After using the dot product in this information fusion step, we search the connected areas within the fused location map. We then calculate the minimum bounding rectangles of the object locations, which completes the information fusion and target object positioning.

In Fig. 4, (a) and (d) are the shallow and the deep class activation maps for a sample image, (b) and (e) are their binary images after the threshold-based processing, and (f) is the result of the dot product operation applied to (b) and (e), i.e., the fused binary image. With the dot product, we can filter away the noise from the shallow binary graph.

3.2. Architecture and training of the HF-RSOL network

We utilize the ResNet34 CNN [58] as backbone network, but remove the last fully connected and Softmax layers. The architecture of the modified network is shown in Fig. 5.

The network consists of a convolutional layer, a maximum pooling layer and four layers of network blocks. Each block consists of several basic modules, which in turn, consist of several network layers. The features detected by the first two block layers are basically edge or corner point features, which cannot be aggregated directly to form semantic information. The existing CAM based WSOD methods, such as [15,17,23,48], mainly use the deep feature maps (the fourth block layer) for generating the class activation maps. This fourth layer is also our DCAM. It exhibits a clear bias and areas with high response correspond to objects and those with low response tend to be background. The characteristics of the third layer are more scattered and include both positive and negative samples, but they are also already useful as hints to detect objects. We use the output of the third block layer of the network as the SCAM, and convert its dimensions to $1 \times H \times H$ using 1×1 convolutions.

As shown in Fig. 5, each basic module is connected to the following module using a skip connection. Via the skip connection, each module extracts features through the convolutional layer. The skip connection will be superimposed with the output of the previous module to ensure that the features of the previous layer will not be lost during network transmission. The specific definition is shown in Eq. (1).

$$x_{l+1} = F(x_l, w_l) + \begin{cases} x_l & \text{if } x_l.shape = x_{l+1}.shape \\ h(x_l) & \text{otherwise, i.e., } x_l.shape \neq x_{l+1}.shape \end{cases} \tag{1}$$

where $h(x_l) = w'_l x_l$, $F(x_l, w_l)$ is the convolution, x_l and x_{l+1} are the input and output of the current module, $shape$ is the matrix dimension, and w_l holds the weights of the lth layer.

If the sizes of x_l and x_{l+1} are different, the feature dimension of x_l is adjusted by 1×1 convolutions, corresponding to the dotted line connection in Fig. 5. If their sizes are the same, they are added directly, as shown by the solid line connection in the figure. With the skip connections, the output of each basic module is cascaded with the output of the previous module, so that the CNNs will not have the problem of a disappearing gradient due to the deepening of the network. Thus, the network performance will not degenerate when the number of layers is increased.

The definitions of the network loss function are shown in Eqs. (2) and (3). For the two-class classification problems, we use the former, the latter is for multi-label classification tasks.

$$loss(output, label) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{C} label(x_{i,j}) \log(output(x_{i,j})) \tag{2}$$

$$\begin{aligned} loss(output, label) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{C} & label(x_{i,j}) \log(1 + exp(-output(x_{i,j})))^{-1} \\ & + (1 - label(x_{i,j})) \log\left[\frac{exp(-output(x_{i,j}))}{1 + exp(-output(x_{i,j}))}\right] \end{aligned} \tag{3}$$

Here, $label$ is the real data label, m is the size of the mini-batch, i represents the ith sample in the mini-batch, C is the number of categories, and $output(x_{i,j})$ represents the predicted probability that the ith sample belongs to category j. We train the network by minimizing this loss function. In the actual training process, we first use the large-scale classification dataset ImageNet [59] to pre-train the ResNet34 model. After the pre-training is completed, we load the network parameters into the modified network and randomly initialize the newly added network modules. Then the fine-tuning training is done with the actual dataset.

The parameters during this training are as follows: the batch size is 16, the weight decay is 0.0005, the momentum is 0.9, the learning rate is 10^{-3}, and the number of iterations is 1000. As optimization algorithm, we select a stochastic gradient descent method where the learning rate is reduced by factor 0.1 every 30 iterations.

3.3. Object location map of the HF-RSOL

There are two steps for calibrating the object position information in the RS test images. The first step is to extract the class activation maps, i.e., the DCAM and SCAM. According to the output of the classification network, we extract the DCAM $M_{d,i}$, where $i \in \{0, 1, 2, \ldots, C\}$ represents the predicted category. Then, the output of the third layer of the network module in Fig. 5 is used as the SCAM. The channel number of the feature map is converted to 1 by 1×1 convolutions, which is

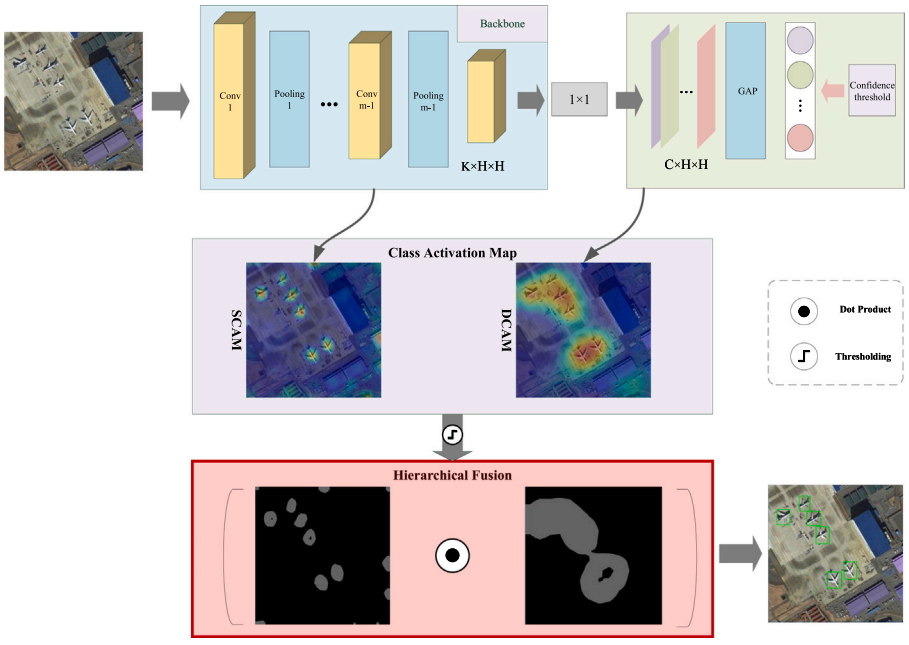


Fig. 3. Network framework of the HF-RSOL.

equivalent to adding the values of each dimension of the feature map. The output result is recorded as the SCAM M_s.

The second step is to fuse the extracted maps to obtain a fused location map to calibrate the object location. The maps $M_{d,i}$ and M_s, which can be considered as grayscale images, therefore need to be converted into binary matrices. We adaptively use the maximum inter-class variance [60] to find the proper threshold values of the foreground and background in the binary maps. By traversing different thresholds, our method calculates the difference between the background and foreground based on the gray values. The larger the inter-class difference is, the larger is also the variance between the foreground and background. The threshold is thus the value at which the inter-class difference reaches its maximum.

For each RS image, both the DCAM and SCAM are segmented separately. Each map of size $M \times N$ is divided into foreground and background based on a threshold T. If a "pixel" has a value larger than T, it is marked as foreground and, otherwise, as background. Let the number of pixels in the foreground be N_f, their average gray value be G_f, the number of pixels of the background be N_b, and their average gray value be G_b. The average gray value of the whole image is G and the class variance between the background and foreground is σ. Then

$$N_f + N_b = M \times N \tag{4}$$

$$G = G_f \frac{N_f}{M \times N} + G_b \times \frac{N_b}{M \times N} \tag{5}$$

$$\sigma = \frac{N_f}{M \times N} \times (G - G_f)^2 + \frac{N_b}{M \times N} \times (G - G_b)^2 \tag{6}$$

By substituting (5) into (6), we obtain the equivalent formula:

$$\sigma = \frac{N_f}{M \times N} \times \frac{N_b}{M \times N} \times (G_f - G_b)^2 \tag{7}$$

If $\sigma(T)$ is the variance between the background and foreground, then:

$$T_{opt} = \text{argmax}(\sigma(T)) \text{ with } T \in \{0, 1, \ldots, 255\} \tag{8}$$

where T_{opt} is the threshold value of the method of maximum inter-class variance. Then, the foreground and background are segmented and binarized using the obtained threshold T_{opt}.

We name the binarized SCAM and the DCAM maps B_s and B_d, respectively. The smallest bounding rectangle of the connected region is obtained as the positioning result of the location map. Finally, we sort the bounding rectangles discovered in B_d from high to low confidence and calculate the intersection over union (IOU) between all rectangular boxes in B_s and B_d. This process is shown in Algorithm 1.

Algorithm 1 Details of the Fusion Location Map Generation Procedure

Require: Binarized SCAM and the DCAM maps B_s and B_d;
Ensure: object bounding boxes res;

```
for i ∈ {1, 2, ..., C} do
    for b_d in B_{d,i} do
        for b_s in B_s do
            iou = IOU(b_s, b_{d,i});
            if (iou ≤ 1 or iou ≥ 0.02) and (b_s.category = null) then
                b_s.category = i;
                res.append (b_s);
            else
                res.append (b_d);
            end if
        end for
    end for
end for
```

$$IOU(b_s, b_{d,i}) = \frac{Size(b_s \cap b_{d,i})}{Size(b_s \cup b_{d,i})} \tag{9}$$

The formula of the IOU is given in Eq. (9), where b_s and $b_{d,i}$ are the object bounding boxes in the location maps B_s and B_d, respectively. By calculating the IOU, we can judge whether the rectangular box in the shallow location map contains the one in the deep location map. According to our experience, the threshold values for the IOU should be set between 0.02 and 1 to ensure that no region containing the object in the shallow location map will be lost.

4. Similarity-based divergent activation for object detection from remote sensing images

For objects of small proportion and dense distribution in RS images, in addition to the inaccurate object location, there is also the challenging problem of detection misses.

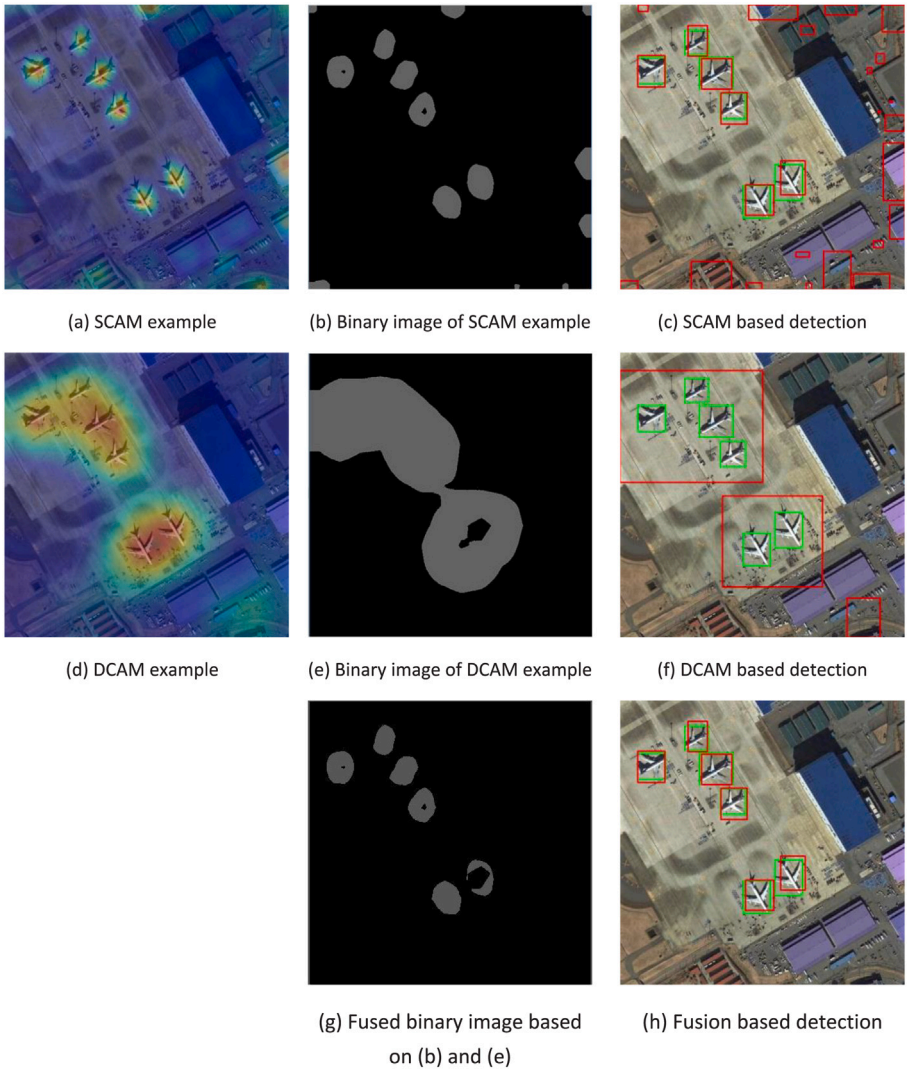


Fig. 4. Examples of SCAM, DCAM and fusion based detection. The ground-truth boxes in (c), (f), and (h) are green and the predicted bounding boxes are red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

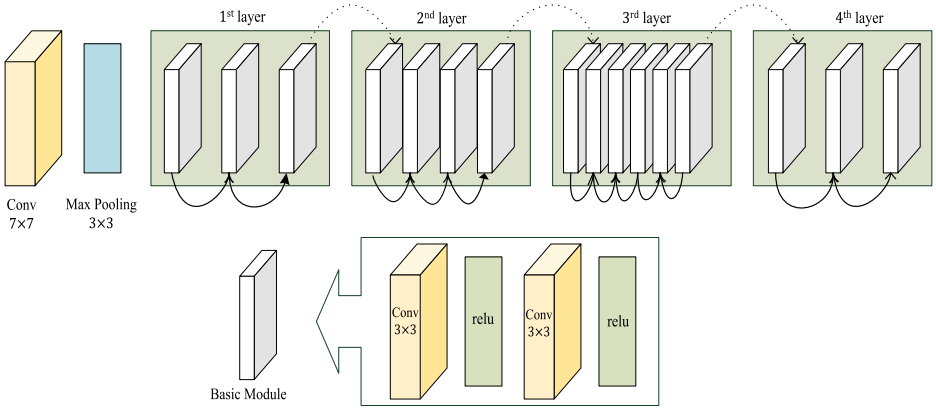


Fig. 5. ResNet34 backbone based network architecture for the HF-RSOL.

The four example images shown in Fig. 7 contain a total of 14 airplanes. However, only 6 airplanes are detected when we use the HF-RSOL method from Section 3, meaning that 8 are lost. By visualizing the shallow and the deep location maps of the image in Fig. 6, we find that the airplanes are not lost in the shallow layer, but their response is weak.

When the fourth layer of the CNN extracts the third layer features and gathers them to form semantic features, it suppresses the areas that are not sufficiently responsive. It only retains the shallow features with higher response, which causes the detection misses.

In order to solve this problem, we propose the novel *Similarity-Divergent Activation Based Remote Sensing Object Detection* (SDA-RSOD)

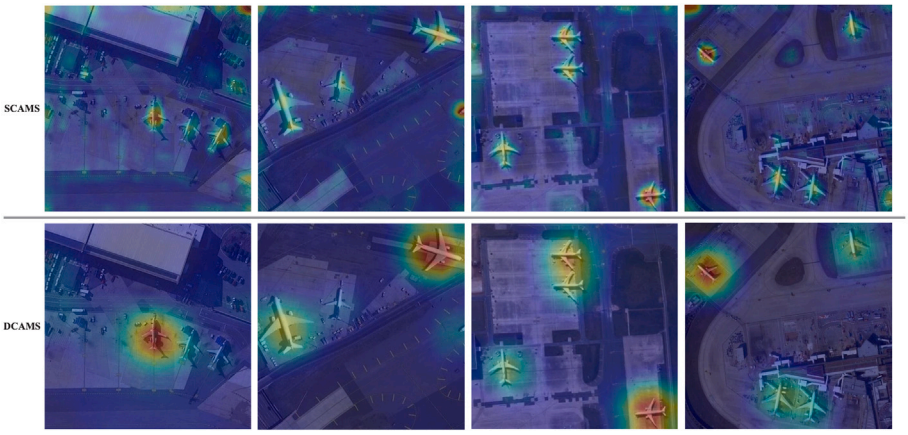


Fig. 6. Examples of shallow and deep class activation maps.

Fig. 7. Examples of detection misses using the HF-RSOL: the ground-truth boxes are green and the predicted bounding boxes from the HF-RSOL are red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

method, which is based on our HF-RSOL. In Section 4.1, we present the framework of the SDA-RSOD and in Section 4.2, we introduce its training mechanism.

4.1. Framework of the SDA-RSOD

The framework of the proposed SDA-RSOD is illustrated in Fig. 8. We essentially add the DA module and the similarity module after the output of the third layer of the basic network of the HF-RSOL.

In the SDA-RSOD, the DA module is used for increasing the response intensity of low-response areas and activating the unresponsive regions in the shallow feature map. In the DA module, the feature dimension of the third layer output of the network is increased from $N \times H \times H$ to $(N \times K) \times H \times H$ through 1×1 convolutions. Then the output of the DA module is used as the input of the similarity module. In the RS field, many of the objects of the same category may appear in a single image. Including a similarity module can therefore be helpful to find the semantic information of these same objects. The similarity module builds a similarity matrix to improve the response intensity of similar regions, to improve the feature distribution, and to suppress the response intensity of background noise regions. Finally, the output of the similarity measurement module is used as the input of the fourth

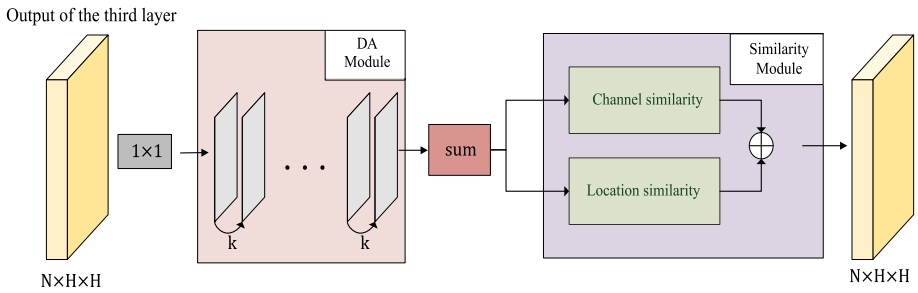


Fig. 8. Similarity and divergent activation based object detection framework.

layer network. Next, we will present a comprehensive analysis of the DA module and the similarity module.

(1) DA Module: When visualizing the shallow and the deep location maps, we find that some of the objects with small proportion and dense distribution have higher response levels and some have lower response levels. In the deep location map, objects with lower response levels will often be lost, resulting in detection misses. Each convolution kernel is equivalent to a feature extractor. The features are extracted by the convolution kernel to form a feature map. By expanding the difference between each dimension of the feature map, more activation regions are obtained. This enhances the target object response.

The purpose of the DA module is to increase the dimension of each one-dimensional feature map output from the CNN. The cosine distance between the K feature maps is enlarged to activate regions, which leads to the formation of more active regions. As shown in Fig. 8, the dimension of the output feature maps M_N of the third-layer network is increased to $(N \times K) \times H \times H$ through 1×1 convolutions, where N is the amount of the feature maps of the third-layer network. We denote the feature map at dimension n at this time using M_n^k, where $k \in \{1, 2, \ldots, K\}$ and $n \in \{1, 2, \ldots, N\}$.

In the initial stage of the dimension n, the feature map $M_n^1 = M_n^2 \cdots = M_n^K$. In order to find more active regions, the response regions of the K feature maps should be different. The cosine distance between each feature map is taken as the function loss constraint, which enforces that the activation regions of each feature map become different in the training of the network. This increases the response area to a certain extent and makes the original response more sensitive. The constraint loss function is defined as (10).

$$\arg\min_{\alpha} L(\alpha) = \sum_{1 \le k < k' \le K} S(M_n^k, M_n^{k'}), \tag{10}$$

Here α holds the parameters of the CNN. $S(M_n^k, M_n^{K'})$ is the cosine distance between M_n^k and $M_n^{K'}$ and its formulation is:

$$S(M_n^k, M_n^{k'}) = \frac{M_n^k \cdot M_n^{k'}}{\|M_n^k\| \cdot \|M_n^{k'}\|}. \tag{11}$$

By minimizing the loss function $L(\alpha)$, the difference of the response regions between the K feature maps is enlarged. This increases the intensity and the amount of the response regions. Finally, through the *sum* module, each feature map M in the dimension n is accumulated and summed to obtain the fused feature map. The *average* module is defined as Eq. (12).

$$M_n = \frac{1}{K} \sum_{0}^{K-1} M_n^i, \tag{12}$$

Here M_n is the output feature map of dimension n. By connecting the feature maps of each dimension, the DA module outputs M'_N of $N \times H \times H$ dimensions.

(2) Similarity Module: After the DA module, some regions are activated, but they are not necessarily all target regions. Performing divergent activation mainly enlarges the cosine distance between the feature maps, but does not consider the characteristics of the target regions. Therefore, the similarity module is introduced to further modify the output of the DA module. It suppresses the response intensity of the non-object regions and improves the response intensity of the target areas.

In the shallow location map, most regions with high response are target areas. Based on this fact, the similarity of features in different channels and the features of the feature maps of the same dimensionality are measured to obtain the final output. We utilize the feature map output by the DA module to construct a channel similarity matrix. The mutual dependence between the channel maps allows us to measure the feature similarity of the regions on each channel. We then compute the similarity of the feature maps in the same dimension and measure the similarity of features at different locations by constructing a location similarity matrix. Through two different similarity constraints, the background noise in the feature map is removed and the response intensity of the target area is improved. Next, we introduce the details of the channel similarity module and the location similarity module, respectively.

Channel Similarity Module: Let the output feature map of the DA module be M, and its dimension be $N \times H \times H$. Since each convolution kernel can be regarded as different feature extractor, each channel can be regarded as the response of specific features. The regions with strong response are distributed in the features of each channel. By using the feature map between different channels, a similarity matrix is constructed to emphasize the interdependence of the channels, so as to improve the representation of features. The flow of the channel similarity module is shown in Fig. 9.

First, we convert the dimension of the feature map M to $(N \times H) \times H$ and denote this new map as A. We calculate the similarity between channels using Eq. (13), where X is the channel similarity matrix constructed to measure the degree of correlation between different channels.

$$X = A \cdot A^T \tag{13}$$

Then, we use the softmax function to normalize the similarity matrix X, as in Eq. (14). Here, $x_{i,j}$ represents the similarity between the feature of channels i and j.

$$x_{i,j} = \frac{\exp(A_i \cdot A_j)}{\sum_{i=0}^{N-1} \exp(A_i \cdot A_j)} \tag{14}$$

Finally, the output E of the channel similarity module is defined in Eq. (15). The element-wise summation operation in X and A measures the similarity between each current feature map A_j and each channel. Then its dimension is readjusted from $(N \times H) \times H$ to $N \times H \times H$ as the output of the similarity module.

$$E_j = \sum_{i=0}^{N-1} (x_{i,j} A_i) + A_j \tag{15}$$

Location Similarity Module: The receptive field of the convolution kernel is usually limited. Therefore, the representation it learns likely ignores the correlation between the features. The location similarity

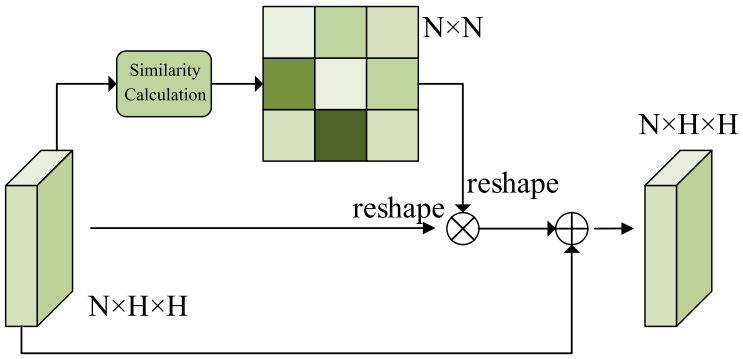


Fig. 9. Channel similarity module.

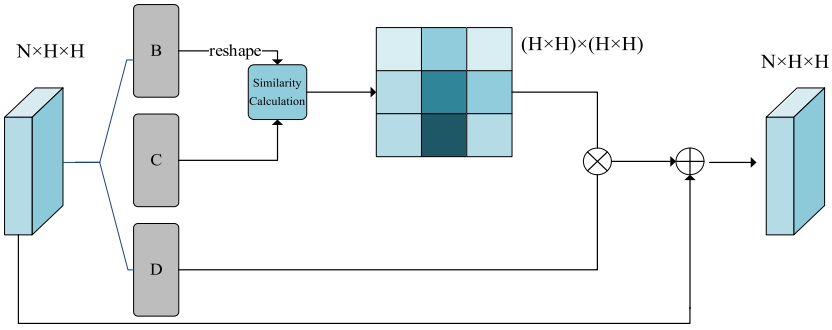


Fig. 10. Location similarity module.

matrix increases the response intensity of similar features, thereby improving the feature representation. The specific process of the location similarity module is shown in Fig. 10.

First, the m-dimensional feature map is transformed to size $H \times H$ through two 1×1 convolutions, which are denoted as B and C, respectively. The similarity between different positions is calculated by Eq. (16). Here, Y is the constructed location similarity matrix. It measures the correlation degree of features in different positions.

$$Y = B^T \cdot C \tag{16}$$

The location similarity matrix is then normalized via the softmax function based on Eq. (17), where $X = H \times H$ and $s_{i,j}$ indicates the similarity between the location features.

$$S_{j,i} = \frac{\exp(B_i \cdot C_j)}{\sum_{i=0} X \exp(B_i \cdot C_j)} \tag{17}$$

Then, the feature map M is processed by a 1×1 convolution to keep its dimension unchanged, and the result is recorded as D. The similarity between the current feature map D_j and each position is measured by the elementwise sum operation in Y and D. Its dimension is readjusted from $N \times (H \times H)$ to $N \times H \times H$. The specific operation is shown in Eq. (18).

$$F_j = \sum_{i=0}^{X} (s_{ji} D_i) + D_j \tag{18}$$

4.2. Training mechanism of the SDA-RSOD network

The training mechanism is basically the same as for the HF-RSOL. The loss function is defined in Eq. (19).

$$L_{SDA-RSOD} = loss(output, label) + \lambda L_{DA}(\alpha), \tag{19}$$

The loss is composed of the classification cross entropy $loss(output, label)$ and the DA constraint component. $loss(output, label)$ is consistent with the loss function in HF-RSOL as defined in Eqs. (2) and (3). The DA constraint component uses the cosine distance to measure the similarity of the feature maps, as defined in Eq. (10).

The hyperparameter λ of the loss function is used for adjusting the response degree of the activation region of the DA module. When λ is large, the response degree will increase. If it is small, the response degree decreases. As can be seen from the previous section, the similarity module is based on the premise of the high response region to represent objects. When λ is too large, it will create a feature imbalance, that is, the high response region does not necessarily correspond to an actual target object. As a result, the overall performance of the network declines and the object detection effect is reduced. In Sections 5.4 and 5.6, we show how to configure λ properly.

5. Experimental results and analysis

We now evaluate our proposed methods with comprehensive experiments on a public dataset and a self-assembled dataset, namely DIOR [9] and WSADD [23,32]. We first describe the datasets and the evaluation criteria used in Section 5.1. We then perform a series of experiments using the HF-RSOL and the SDA-RSOD and compare their performance with state-of-the-art object detection methods in Section 5.2. The question of how the location maps of different layers influence the final performance is investigated in Section 5.3. In Section 5.4, we verify the utility of the DA module and the similarity module. The impact of the hyperparameters on the performance of SDA-RSOD is analyzed in Section 5.6. As environment for all experiments, we use a Ubuntu 16.04 PC with an Intel Core i7-7700 processor with 32 GB RAM and an Nvidia GTX 1080Ti graphics card as well as Anaconda3-2019.03 and Pycham Professional 2018.

5.1. Description of the datasets and evaluation criteria

The *DIOR Dataset* [9] is a high-resolution RS image dataset used for object detection. It is an extension of the NWPU-VHR10 dataset [61] and contains 192,472 object instances on 23,463 RS images with

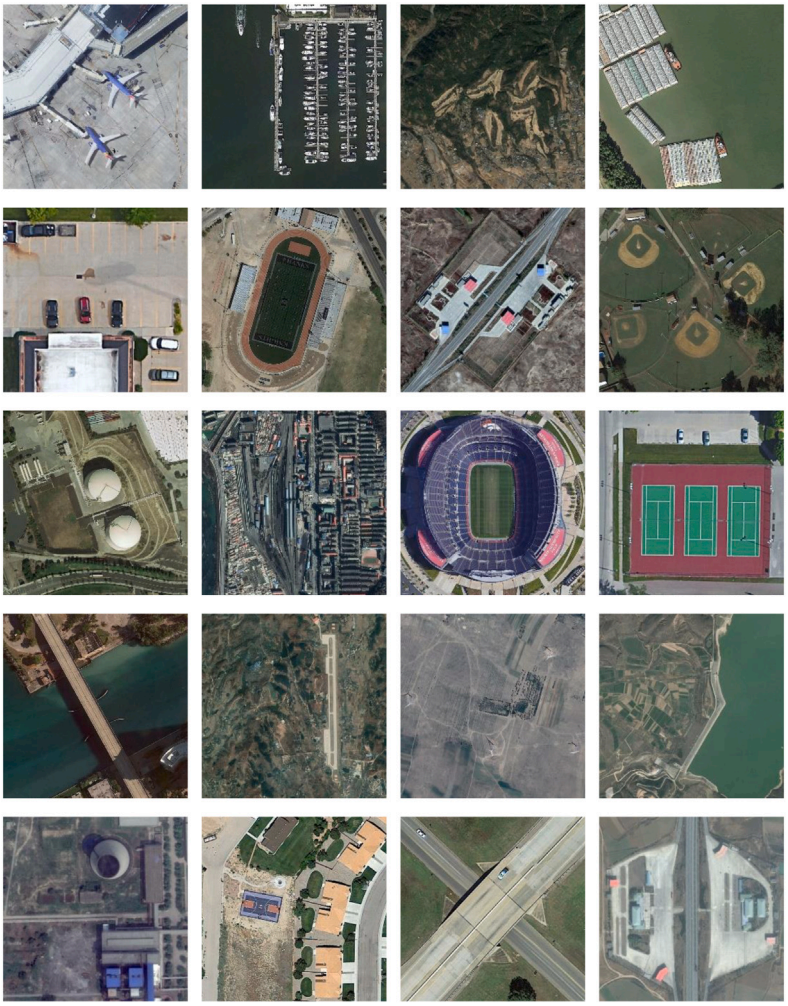


Fig. 11. Some examples taken from the DIOR dataset [9].

a varying spatial resolution of 0.5 to 30 m. There are 20 types of objects, such as airplanes, airports, baseball fields, basketball courts, bridges, chimneys, dams, highway service areas, highway toll stations, ports, and golf courses. All images are collected from Google Earth satellite imagery and the object annotations were performed manually by human professionals. The image resolution span is large and many of the targets, such as ships, airplanes, cars, and chimneys, are small. The example images shown in Fig. 11 contain various typical remote sensing image objects. For this dataset, the training set contains 11,725 images and the test set the remaining 11,738.

The WSADD dataset was made public in our recent work [23]. It comprises 700 RS images in total, 400 of which contain an airplane (the "positive sample set") and the other 300 do not (the "negative sample set"). In the process of dataset construction, the spatial resolution of the images was controlled between 0.3 m and 2 m, and the size was fixed to 768×768 pixels. We sought to collect images from different sensors during different daytimes, different seasons, and different light intensities to ensure that the dataset has a high diversity. Fig. 12 shows some examples images from WSADD. Here, the training set contains 600 images and the test set contains 100.

The DIOR dataset provides RS images with annotations for a wide range of objects. This allows us to test the ability to detect objects with a variety of different features. WSADD is designed for one specific, important application: the detection of airplanes. Aircraft are small compared to the image size and often appear densely distributed. By using both datasets, we can investigate the ability to find and distinguish closely co-located objects but also test the general detection capabilities and thus prevent overfitting to a specific application scenario.

5.1.1. Evaluation criteria

Recall and Precision are the two basic performance metrics for object detection. They are defined as follows:

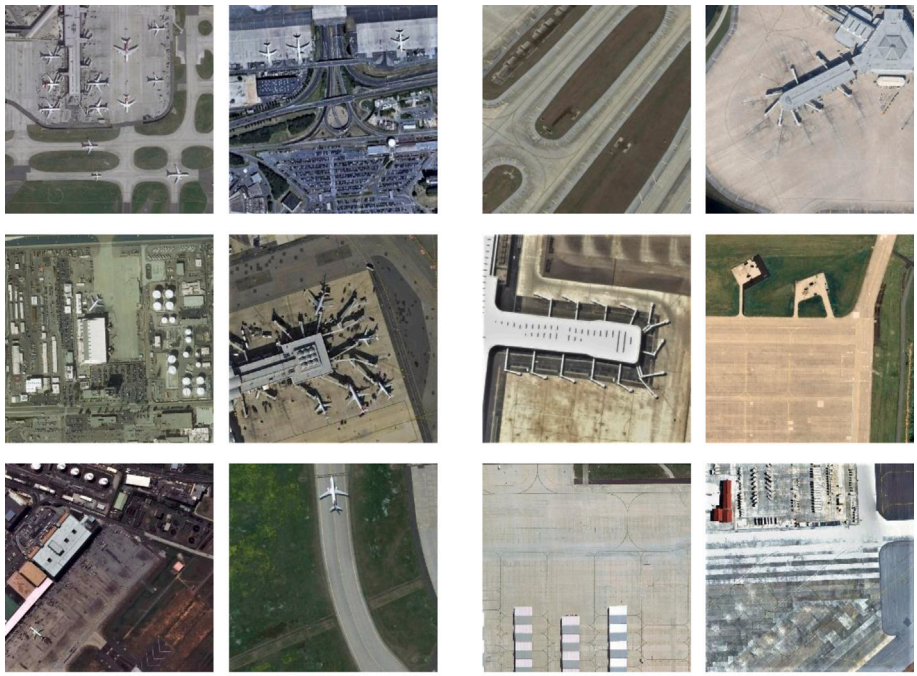
$$Recall = \frac{TP}{TP + FN} \tag{20}$$

$$Precision = \frac{TP}{TP + FP} \tag{21}$$

Here, TP, the number of true positives, corresponds to the correctly detected objects of the target class. If an object instance is detected as the right class and the proposed bounding box has an intersection overlap (IOU, see Eq. (9)) with the ground truth bounding box greater than 0.5, it is a true positive. The number FP of false positives (or false alarms) counts the objects incorrectly assigned to the class by the detection stage. A detection is a false alarm if no corresponding object belonging to the same class is found in the test data with an intersection overlap greater than 0.5. FN is the number of the undetected or misclassified objects of the class (the false negatives).

For the WSADD dataset, there are only two categories. The performance of a detector hence can directly be evaluated by the recall and the precision. The higher the recall rate, the greater the proportion of all objects that are correctly detected. The higher the precision, the greater the proportion of correct detections in all detection results.

As the DIOR dataset contains multiple categories of objects. It thus is difficult to effectively evaluate the performance of the algorithms using recall or precision alone. We therefore use the *mean Average Precision* (mAP) metric to measure the object detection accuracy of



(a) Positive images (b) Negative images

Fig. 12. Example images from the WSADD dataset [23].

Table 1
Backbone architectures of the alternative methods.

Method	Backbone	Model size (MB)	Million multi-Adds	Million parameters
CAM	*ResNet* [58]	> 80	4000	63.5
ACoL	VggNet [64]	500	15 300	138
AlexNet-WSL	AlexNet [51]	> 200	720	60
DANet	VggNet [64]	500	15 300	138
Ours	ResNet [58]	> 80	4000	63.5

Table 2
Comparison with alternative methods on WSADD.

Method	TP (number)	FP (number)	FN (number)	Precision (%)	Recall (%)
CAM [15]	30	170	278	15.00	9.74
ACoL [17]	23	153	285	13.07	7.47
AlexNet-WSL [23]	33	167	275	16.50	10.71
DANet [48]	23	160	285	12.57	7.47
HF-RSOL	251	137	57	64.69	81.49
SDA-RSOD	**280**	**70**	**28**	**80.00**	**90.91**

different recall rates. We refer the readers to [62] for more details about the mAP.

5.2. Comparison experiments with state-of-the-art

In order to verify the effectiveness of our weakly supervised methods, we compare them with four very recent state-of-the-art algorithms, namely CAM [15], AlexNet-WSL [23], DANet [48], and ACoL [17]. These methods pursue weakly supervised learning of class localization maps from the CNN feature maps for object detection. While CAM, DANet, and ACoL are known for their good performance on natural scenes, AlexNet-WSL has been designed specifically for airplane detection (and thus will show good results on the WSADD dataset used in our experiments here as well).

Table 1 shows the backbone architectures of the investigated methods. The algorithmic complexity of training and testing such models can roughly be gauged by its *compactness*, i.e., the model size and the number of parameters, and *efficiency*, which is measured in terms of the number of Multi-Adds [63]. From the model structure, we can conclude that our approach is more compact than DANet and ACoL and should thus be faster in object detection.

The detection results of all methods on the WSADD are shown in Table 2. DANet and ACoL were developed with the problem of inaccurate object positioning in mind. In natural scenes, target objects usually account for large proportions of the detected images. There, the solution is mainly to expand the response region to improve the positioning. However, in RS scenes, this idea performs worse than the CAM, which shows that methods designed for natural scenes are not suitable for RS images. Still, even the AlexNet-WSL method developed specifically for RS images cannot achieve acceptable positioning and the number of correctly detected objects is only 33 out of 308.

Our HF-RSOL marks an obvious improvement in terms of precision and recall. Without increasing the FP rate, the number of TPs has risen significantly to 251, i.e., to 7.6 times the highest TP rate of any of the compared methods. The information fusion step therefore has a very strong positive impact and improves the performance by almost an order of magnitude. After adding the DA and similarity modules, the overall detection performance of the resulting SDA-RSOD marks another significant improvement. The precision increases from 0.65 to 0.80 and the recall rate further improves from 0.81 to 0.91. We can now detect 280 objects. The number of false positives is now 70, whereas the smallest FP value of any compared approach is 153. In other words, we can now detect almost 8.5 times as many objects and have a FP value of less than half of what the state-of-the-art CAM-based methods deliver.

SDA-RSOD builds on the information fusion implemented in HF-RSOL. It can further increase the number of detected targets, indicating that it can alleviate the problem of detection misses of objects with small proportion and dense distribution. The number FP of false positives also decreases, which shows that the modules introduced in the SDA-RSOD can improve the CNN feature map and also improve the accuracy of the deep location map. This confirms that using the deep

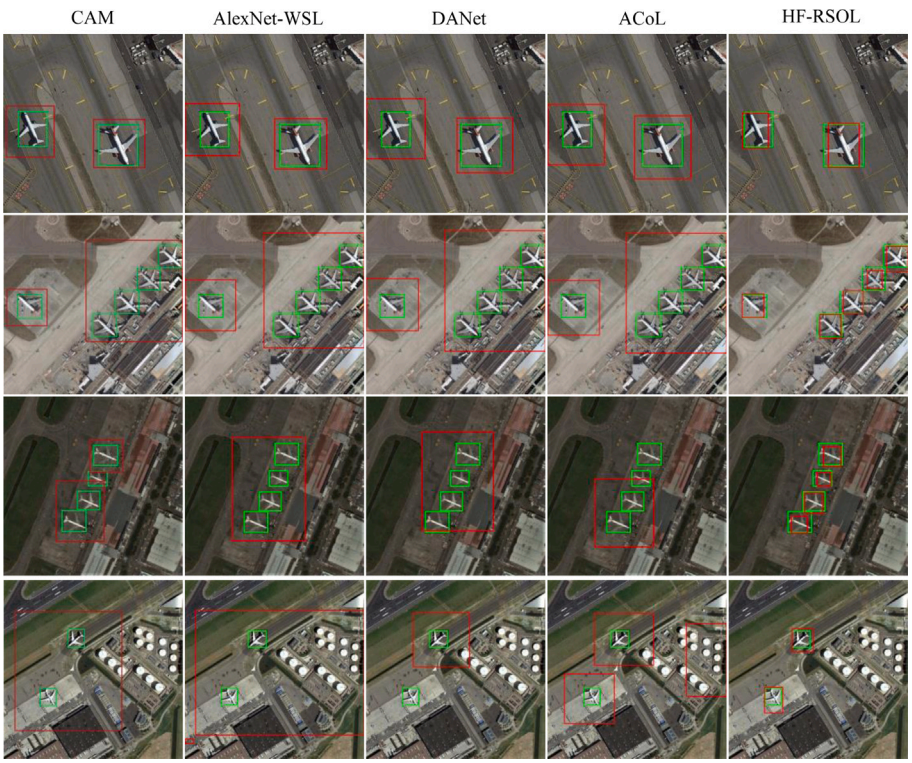


Fig. 13. Examples of detection results by different object location methods on WSADD. The ground-truth boxes are green and the predicted bounding boxes from the different object location methods are red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

location map to filter the positioning results in the shallow location map leads to fewer false alarms.

In Fig. 13, the green boxes are the true object positions and the red boxes are the detection results. We see that the object positioning accuracy of the HF-RSOL is better compared with CAM, AlexNet-WSL, DANet, and ACoL. From the first row of images we find that all methods provide accurate positions for relatively large objects. However, when the proportion of the objects becomes smaller, as in the other three rows, the positioning effect of the CAM, AlexNet-WSL, DANet, and ACoL is poor. The figure further shows that typically, the proportions of the objects in RS image are small and their distribution is dense, which likely caused the object positioning inaccuracies.

Table 3 provides the detection results on the DIOR dataset in comparison with several other state-of-the-art methods. Both the HF-RSOL and the SDA-RSOD outperform the other CAM algorithms on almost every single class. The only exceptions are the class bridges, where DANet is better than HF-RSOL, and the classes stadiums and train stations, where it also outperforms SDA-RSOD. For airplanes and tennis courts, our methods improve the detection accuracies by more than 10% in terms of mAP. They can also correctly detect at least four times as many airplanes, ships, storage tanks, and vehicles than any of the approaches used for comparison — including AlexNet-WSL, which was specifically designed for airplane detection.

From the results of the AlexNet-WSL and the HF-RSOL, we could argue that many objects are not detected due to the inaccurate positioning. For example, the average precision of the AlexNet-WSL method is less than 3% for airplanes, less than 8% for tennis courts, and less than 10% for wind mills. The mAP of the HF-RSOL here is already 14.33%, 18.00%, and 14.62%, respectively. The SDA-RSOD further improves it to 19.51%, 26.52%, and 19.16%, respectively.

For storage tanks, ships, and vehicles, the improvement is large, but the positioning effect is not yet sufficient. The object detection of the HF-RSOL is mainly based on the difference between the categories. Storage tanks, ships, and vehicles often appear in clusters. The problem of inaccurate positioning of such targets is mainly due to this dense distribution. Storage tanks, for example, are often concentrated in a tight group, and it is nearly impossible to accurately locate each separate storage tank.

By adding the similarity module and the DA module to HF-RSOL, we obtain the SDA-RSOD and the mAP value increases by another 2.4%. The average precision (AP value) of large objects, such as train stations and airports, remains at approximately the original level. However, for objects with small proportion and dense distribution, such as airplanes and tennis courts, the mAP has been greatly improved, which shows the effectiveness of the SDA-RSOD. Only in three categories, namely expressway toll stations, overpasses, and stadiums, SDA-RSOD performs slightly worse than HF-RSOL. The difference is less than 1% in each category.

We also compare our approaches with the three state-of-the-art multi-instance learning methods WSDDN [37], OICR [39], and PCIR [45]. Their backbones are all the VggNet [64], which has more variables and is thus harder to train compared to ResNet 34 used in our methods (see Table 1). Still, in terms of the mAP, our methods have obvious advantages over WSDDN [37] and OICR [39], while being less than 1% worse than PCIR [45], which is the most recent approach and was published in 2020. Still, our SDA-RSOD has a better precision than PCIR on 11 of the 20 classes of the dataset.

With regard to the average precision of each category, our methods have the overall best stability, whereas the MIL-based methods perform sometimes very well and in some categories badly: PCIR has an mAP of 24.92%, but its geometric mean average precision (GMAP) [65,66] is only 13.58%. Our HF-RSOL already has a better GMAP of 18.2% and the SDA-RSOD even reaches 20.91% (at an mAP of 24.11%). These are the best results over all compared methods, which emphasizes the better robustness of our methods over different classes.

The training of MIL-based methods mainly follows a two-stage approach. They first decompose images into a series of proposals and then iteratively select the most contributing proposal as the pseudo instance-level label to train object detectors under multiple instance learning constraints. This is what makes them very suitable for object detection

Table 3
Detection average precision (%) of alternative methods on DIOR.

Class	Method								
	Multi-instance learning			Class Activation Map (CAM) methods					
	WSDDN [37]	OICR [39]	PCIR [45]	CAM [15]	AlexNet -WSL [23]	DANet [48]	ACoL [17]	HF-RSOL	SDA-RSOD
Airplane	9.06	8.70	**30.37**	2.66	2.94	1.33	0.15	14.33	19.51
Airport	**39.68**	28.26	36.06	34.63	35.38	33.41	7.62	36.84	38.86
Baseball field	37.81	44.05	**54.22**	16.87	17.92	13.46	2.38	20.31	26.40
Basketball court	20.16	18.22	**26.60**	16.70	18.20	17.95	0.00	18.33	23.56
Bridge	0.25	1.30	9.09	10.59	12.10	12.99	0.00	12.80	**13.30**
Chimney	12.18	20.15	**58.59**	25.43	25.91	21.60	0.04	26.51	26.84
Dam	0.57	0.09	0.22	17.66	18.71	17.20	6.10	22.22	**25.33**
Expressway service area	0.65	0.65	9.65	25.40	26.44	25.84	7.82	26.58	**27.09**
Expressway toll station	11.88	29.89	**36.18**	25.87	25.46	19.68	0.78	27.18	27.17
Golf course	4.90	13.80	32.59	56.23	56.56	53.98	27.72	58.30	**61.04**
Ground track field	42.35	57.39	**58.51**	17.68	19.24	19.86	13.18	20.43	20.89
Harbor	4.66	10.66	8.60	12.86	12.91	12.63	9.43	14.66	**16.78**
Overpass	1.06	11.06	21.63	25.32	25.83	24.31	20.56	**26.29**	25.57
Ship	0.70	9.09	**12.09**	0.11	0.64	0.43	0.13	5.78	8.28
Stadium	63.03	59.29	**64.28**	9.27	10.39	12.37	0.00	11.15	10.34
Storage tank	3.95	7.10	**9.09**	0.84	1.19	0.56	0.63	5.64	7.53
Tennis court	6.06	0.68	13.62	6.64	7.05	5.85	2.27	18.00	**26.52**
Train station	0.51	0.14	0.30	46.00	47.07	**49.57**	18.68	47.27	48.81
Vehicle	4.55	9.09	9.09	1.48	1.74	1.11	0.17	7.21	**9.28**
Wind mill	1.14	0.41	7.52	9.08	9.78	3.08	0.27	14.62	**19.16**
mAP	13.26	16.50	**24.92**	18.07	18.78	17.37	5.89	21.72	24.11

based on weakly supervised learning. At the same time, this step-by-step training is also more complicated than the end-to-end structure offered by CAM-based methods. In terms of relatively small or stadium-class objects, such as airplane, ship, storage tank, stadium, etc., the MIL based methods provide better detection, which is largely due to the high-quality proposals. For relatively large objects, such as bridges, overpasses, dams, etc., the performance of the CAM based method is better.

In addition, for different scenarios, various optimization strategies need to be used, and the adaptability of MIL algorithms is very limited. Class activation map methods are weakly supervised learning algorithms that locate target objects based on the distinction between classes. Compared with multi-instance learning, their end-to-end network structure is easier to train and transfer.

In Figs. 14 and 15, we compare the behaviors of HF-RSOL and SDA-RSOD on different object classes. We plot the P-R curves for the eight object types that cover small areas in RS images from the DIOR dataset, namely airplanes, storage tanks, baseball fields, tennis courts, basketball courts, vehicles, ships, and windmills. For aircraft and tennis courts, the AP values increased from 14.33% and 20.31% to 19.51% and 26.40%, respectively. The recall rate for aircraft rose from 0.20% to 0.25%, indicating that some of the originally missed aircraft are now correctly detected. The accuracy does not change significantly under the premise of increasing the recall rate. The recall for tennis courts increased from 0.25% to 0.35% while the accuracy remained the same. Densely distributed target types such as, cars, windmill, ships and oil depots, the AP values have somewhat improved and, at the same precision, recall rates have improved.

For basketball courts and baseball fields, there is tangible impact of the spatial resolution and these two types of objects are not always small compared to the image size. The AP values for both categories show a large increase, from 18.33% and 18.00% to 23.56% and 26.52%, respectively. This indicates that for the proper spatial resolutions, SDA-RSOD can improve the detection. The recall rates increased from 0.2% and 0.25% to 0.45% and 0.5%, respectively. However, the increase in recall rate here results in a decrease in accuracy, which indicates that SDA-RSOD is not yet an ideal solution for objects covering a large space in the images.

These results demonstrate the high effectiveness of our methods. Still, while our approaches mark a significant progress, the performance needs to be improved further. The main obstacles are: (1) The lack of instance-level supervision together with the co-location of targets and complicated backgrounds leads to misclassifications. (2) Some objects, such as bridges and roads, have very similar appearance. Thus, while we have clearly advanced the state-of-the-art, the results also show that more improvements are needed for a reliable real-world application.

Let us now compare the SDA-RSOD and the HF-RSOL in more detail. In Fig. 16, the left two columns are the results of the HF-RSOL, and the right two columns are the results of the SDA-RSOD. We see that the originally lost airplanes can be correctly detected by the SDA-RSOD. The HF-RSOL usually only focuses on partial activation response regions, so it overlooks airplanes distributed in other regions. Due to the added DA module, the SDA-RSOD can activate more regions which could contain objects in the shallow feature map. This alleviates the problem of the loss of objects with small proportion and dense distribution.

5.3. Separate experiments on shallow and fused location map

The existing state-of-the-art algorithms usually use deep maps based location methods. In the previous section, we presented and analyzed the performance of such algorithms in comparison to our new methods. In order to further investigate the effectiveness of the object location method proposed in this article, we carry out another set of experiments on the WSADD. We apply the comparison methods using shallow location maps and fused location maps, respectively.

From the experimental results given in Table 4, we find that the precision of the positioning results significantly improves when the fused location map is used, but the recall rate decreases. We also see that in the detection results of the fused location map, the number of correctly detected targets (TP value) has decreased compared with the shallow location map, which leads to a decrease in the recall rate. However, the number FP of false alarms in the shallow location map is very large, resulting in a very low precision. There are significantly fewer false alarms in the fused location map, which improves the precision.

Fig. 17 visualizes the positioning results obtained with the shallow and the fused location map. There are many false alarm frames in the results obtained from the shallow location map. This shows that the shallow location map contains a lot of noise. Notice that HF-RSOL is equivalent to using the Fusion Location Map of CAM as both use the

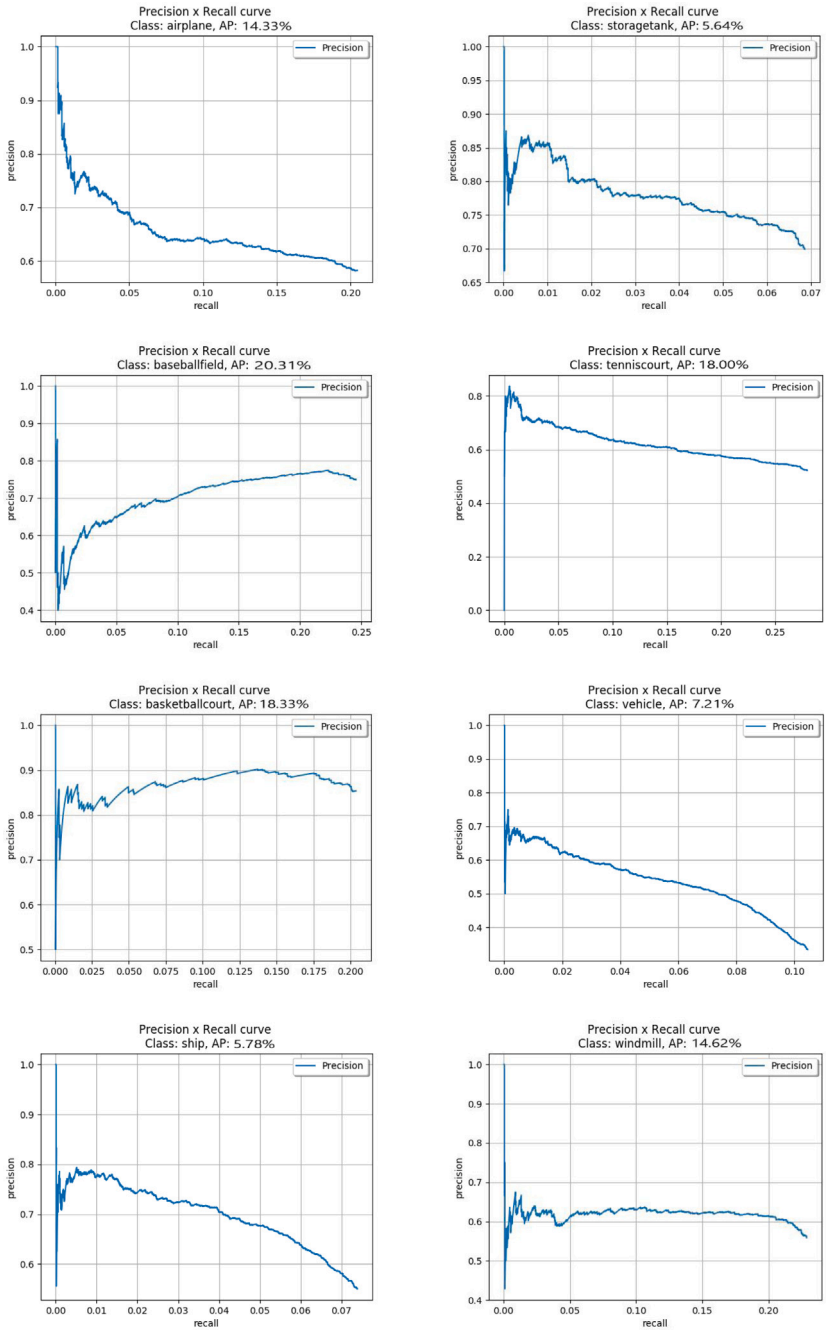


Fig. 14. P-R curves of the HF-RSOL.

Table 4
Experiments with alternative methods on WSADD.

Methods	TP (number)	FP (number)	FN (number)	Precision (%)	Recall (%)
CAM(S)	267	1306	41	16.97	86.69
AlexNet-WSL(S)	260	1256	48	17.15	84.42
DANet(S)	253	2083	55	10.83	82.14
ACoL(S)	260	1100	48	19.12	84.42
CAM(F)	251	137	57	64.69	81.49
AlexNet-WSL(F)	246	147	62	62.60	79.87
DANet(F)	243	151	65	61.68	78.90
ACoL(F)	241	138	67	63.59	78.25

[1] S means "Shallow Location Map", [2] F means "Fused Location Map".

same backbone. The pure CAM method only uses the deep location map.

However, in the deep location map, most of the areas in the location region contain the target objects. There is almost no overlap with the false alarm frames in the shallow location map. Therefore, by calculating the overlap ratio of the positioning box in the shallow and the deep location map, the targets can be accurately located. The fusion of candidate boxes is carried out by Algorithm 1. Compared with the direct use of the shallow location map, the fusion result will have slightly more object losses, but most of the noise is removed.

5.4. Impact and hyperparameters of the DA module

In the DA module, the dimension of the feature map is increased from $N \times H \times H$ to $(N \times K) \times H \times H$. This raises the question how

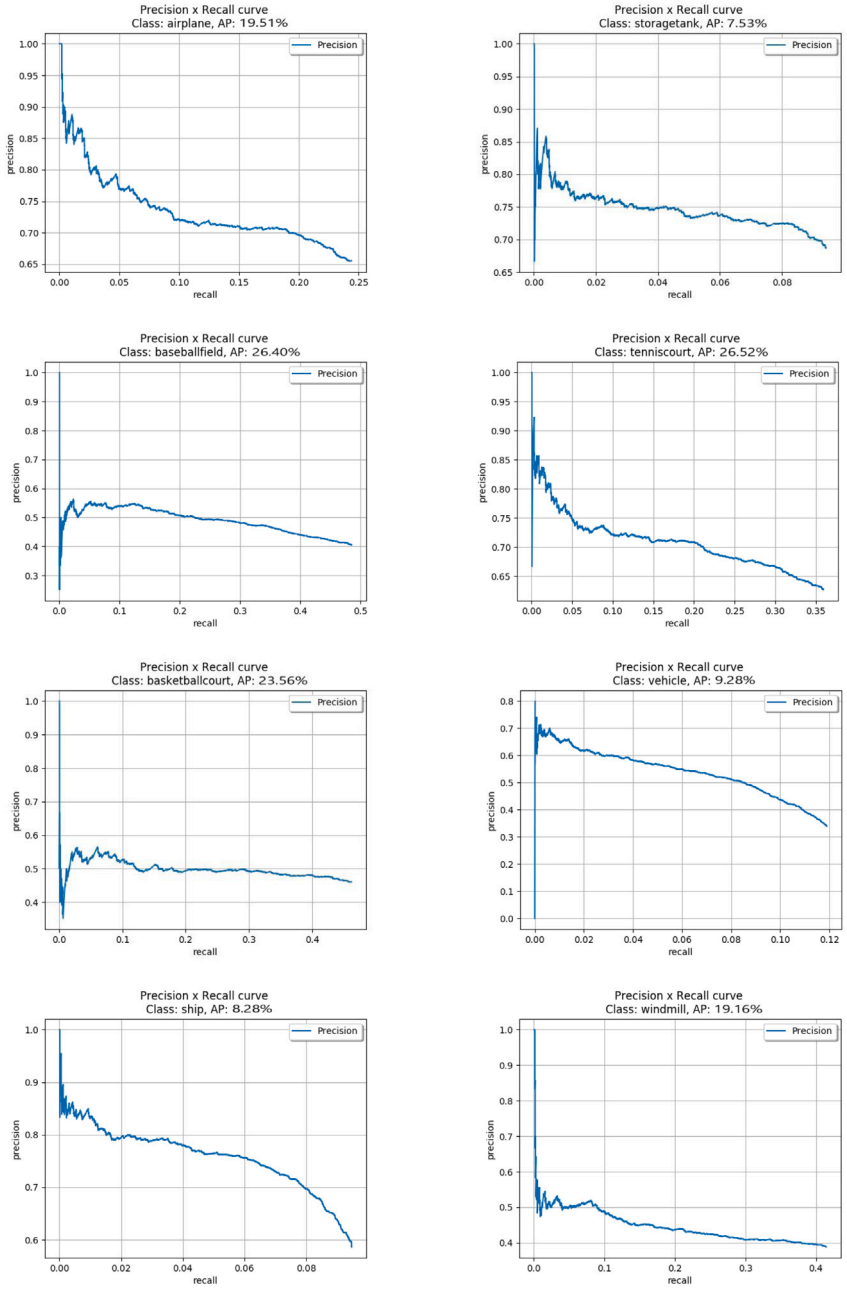


Fig. 15. P-R curves of the SDA-RSOD.

different values of K affect the performance. In the network training phase, the loss function $L(\alpha)$ is constrained by the hyperparameter λ. It is also necessary to analyze the influence of different values of λ. We now conduct an experiment investigating the impact of these two hyperparameters of the DA module.

First, we keep the value of K unchanged and choose different values for λ. We then hold λ constant and use different values of K. We again use the WSADD dataset. The results are shown in Table 5.

When the values of hyperparameters K and λ are both low, the number TP of correctly detected targets is high, which improves the recall rate compared with the fused location map in the HF-RSOL. If we increase the hyperparameters K and λ, the TP value increases only slowly. For $K > 4$, a value of $\lambda \geq 0.1$ begins to lead to a lower TP. If λ reaches 0.5, the network cannot complete the classification task anymore for any K, and incorrect object categories are predicted. As a result, the response region in the extracted deep feature map is not the target object position and the object detection fails. This means that during training, the network should focus on the classification loss. The added DA loss from Eq. (10) cannot be emphasized too much, otherwise it will affect the classification performance of the network.

Further analysis shows that the activation intensity of the DA module increases slowly when the hyperparameter K keeps increasing and λ remains low (around 0.01 and 0.02). Then, the number of correctly detected objects increases continuously. When K remains unchanged and the value of λ is low, the activation intensity of the DA module is moderate, which increases the TP value.

When K is large, say 16, and the λ value is low, the TP value is increased. However, as stated before, when the λ value reaches 0.1 for larger K, the DA module activation is high and the performance degenerates. Compared with the results of the fused location map, the recall rate has hardly improved, but the precision rate has dropped significantly.

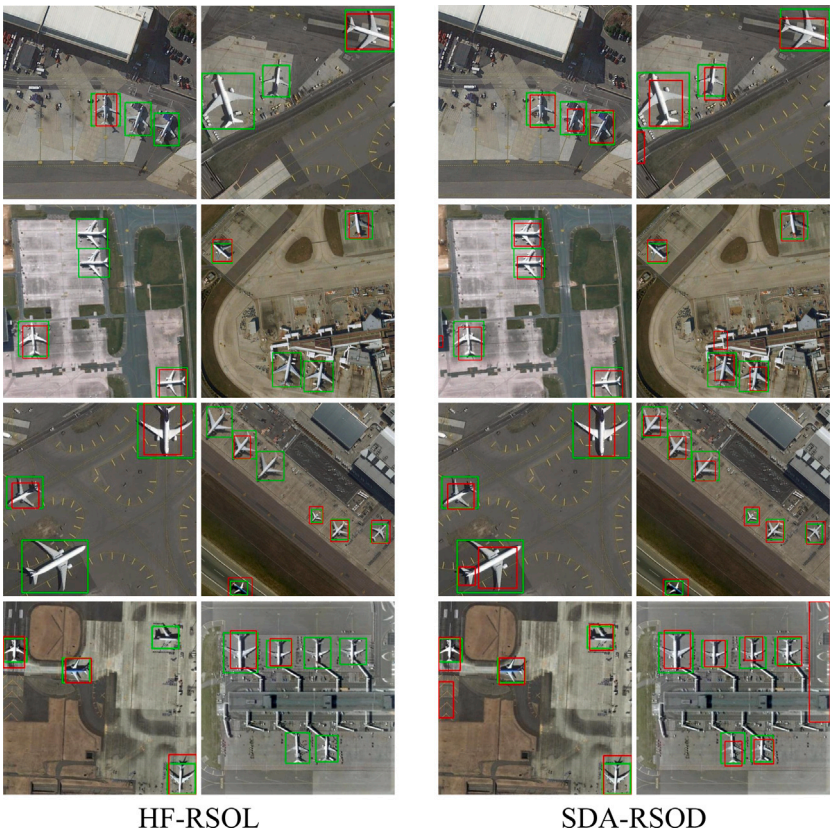


Fig. 16. Examples of detection results by the HF-RSOL and the SDA-RSOD. The ground-truth boxes are green and the predicted bounding boxes from the HF-RSOL and the SDA-RSOD are red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

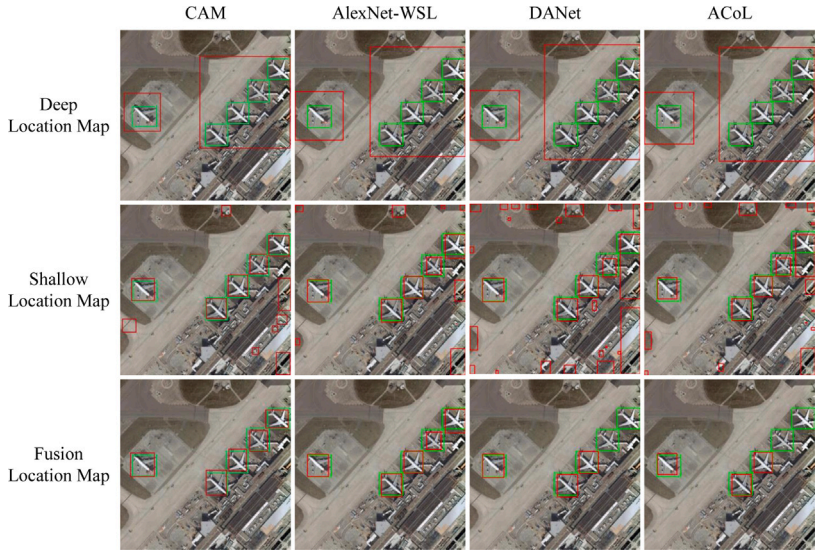


Fig. 17. Object location effect of different location maps. The ground-truth boxes are green and the predicted bounding boxes from the deep, shallow and fusion location maps are red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In conclusion, the addition of the DA module for the SDA-RSOD can increase the number of correctly detected objects and improve the recall rate, but at the same time, it will lead to the increase of the false alarm rate and a decrease of precision. The values of the hyperparameters K and λ affect the activation strength of the DA module. When it gets too high, the performance degenerates. Therefore, the values of K and λ should be set in a lower range.

5.5. Impact of the similarity module

We now investigate whether the shallow feature maps that have not been processed by the DA module can improve the response strength for similar objects and improve the detection performance through the similarity module. We therefore remove the DA module from the basic network and verify the effect of the similarity module separately. The experiment is again performed on the WSADD dataset and its results are shown in Table 6.

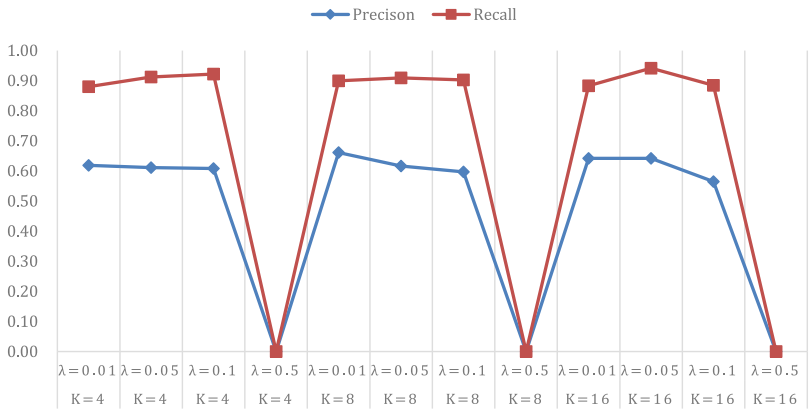


Fig. 18. Precision and recall at different values of K and λ on WSADD.

Table 5
TP, FP, and FN at different values of K and λ on WSADD. See also Fig. 18.

K, λ	TP (number)	FP (number)	FN (number)
$K = 4, \lambda = 0.01$	271	167	37
$K = 4, \lambda = 0.05$	281	179	27
$K = 4, \lambda = 0.1$	284	183	24
$K = 4, \lambda = 0.5$	0	100	308
$K = 8, \lambda = 0.01$	277	142	31
$K = 8, \lambda = 0.05$	280	174	28
$K = 8, \lambda = 0.1$	278	188	30
$K = 8, \lambda = 0.5$	0	100	308
$K = 16, \lambda = 0.01$	272	152	36
$K = 16, \lambda = 0.05$	290	162	18
$K = 16, \lambda = 0.1$	252	194	56
$K = 16, \lambda = 0.5$	0	100	308
HF-RSOL	251	137	57

Table 6
Experimental results to analyze the impact of the similarity module on WSADD.

Methods	TP (number)	FP (number)	FN (number)	Precision (%)	Recall (%)
HF-RSOL	251	137	57	64.69	81.49
SDA-RSOD	280	70	28	80.00	90.91
$C + L$	245	110	63	69.01	79.55
C	239	130	69	64.77	77.60
L	243	126	65	65.85	78.90

[1] C means "Channel Similarity", [2] L means "Location Similarity".

The TP value slightly decreases when the similarity module is added after the shallow feature map of the basic network. In other words, the number of correctly detected objects is reduced compared with the HF-RSOL. At the same time, the number FP of false alarms is also reduced, which shows that the similarity module can enhance the response intensity of similar features and improve the distribution of features. Thus, the fourth layer of the basic network can more accurately locate the regions containing targets while ignoring those without objects. However, some areas with low response will also be suppressed, resulting in a decrease in the recall rate.

Both the channel similarity module and the location similarity module can somewhat reduce the number of false alarms, but the effect is not very strong. This shows that the features are scattered and sparse in the shallow feature map. Only focusing on the feature similarity between the feature maps of the same dimension or only focusing on the similarity of the features on the same channel cannot improve the response strength of similar features. By combining the two similarities, the number of false alarms decreases significantly, which improves the distribution of the features and the response intensity of similar features.

Table 7
TP, FP and FN of "Similarity Module + Different K and λ Values" on WSADD. See also Fig. 19.

K, λ	TP (number)	FP (number)	FN (number)
$K = 4, \lambda = 0.01$	260	86	48
$K = 4, \lambda = 0.05$	272	82	36
$K = 4, \lambda = 0.1$	280	70	28
$K = 4, \lambda = 0.5$	273	88	35
$K = 8, \lambda = 0.01$	275	73	33
$K = 8, \lambda = 0.05$	281	88	27
$K = 8, \lambda = 0.1$	271	94	37
$K = 8, \lambda = 0.5$	266	97	42
$K = 16, \lambda = 0.01$	278	90	30
$K = 16, \lambda = 0.05$	279	101	29
$K = 16, \lambda = 0.1$	277	100	31
$K = 16, \lambda = 0.5$	279	98	29
HF-RSOL	251	137	57

Only using either the channel similarity module or the location similarity module alone is not sufficient. Their combination, however, can effectively improve the response intensity of similar features, improve the feature distribution, and reduce the number of false alarms at the same time.

5.6. Factors influencing the DA and similarity modules

In the above two sections, we have verified the functions of both the DA and the similarity module. We find that the DA module can activate the target areas and improve the recall rate to a certain extent, but also causes more false alarms and a greater decrease in precision. The similarity module can improve the response strength of similar areas, improve the distribution of features, reduce false alarms, and improve accuracy. However, some objects will likely be lost in this process, and some areas with low response in the shallow feature map will be suppressed after passing the similarity module.

In this section, we investigate the combined effect of the DA and the similarity module by again adjusting the activation intensity of the DA module through the hyperparameters K and λ. The experimental results are shown in Table 7. When the hyperparameters K and λ are in the appropriate range, such as $K = 4$ and $\lambda = 0.1$, the DA module can activate and increase the response intensity of the target area and the similarity module can improve the feature distribution. Then, the features in the shallow feature map will not be lost due to low response strength after passing through the fourth layer of the basic network. Furthermore, while the precision remains stable, the recall is improved.

When the values of the hyperparameters K and λ are small, the activation strength of the DA module is too weak to activate more regions and improve the response of the target areas. Then, adding

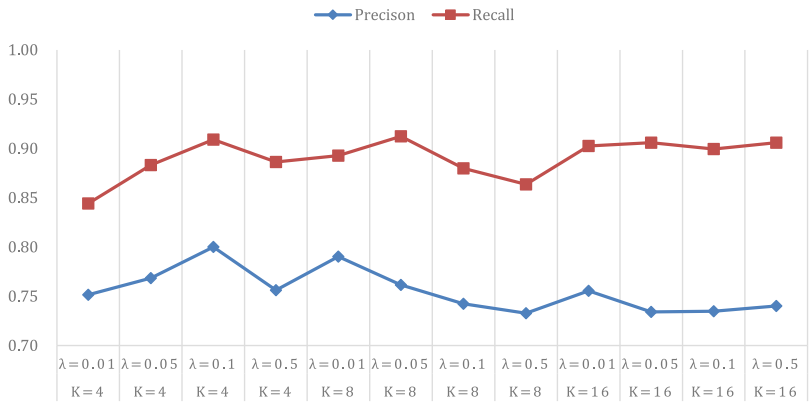


Fig. 19. Precision and recall of "Similarity Module + Different K and λ Values" on WSADD.

the similarity module does not significantly improve the accuracy. The overall performance is not improved and the area activated by the DA module is suppressed by the similarity module. When the values of K and λ are too large, say $K = 16$ and $\lambda = 0.5$, the DA module has a strong activation intensity and the activation area contains more noise. Through the similarity module, the noise can be suppressed to a certain extent, but the number FP of false alarms still increases significantly, resulting in a decrease of the precision. The similarity module can suppress some but not all noise.

When the values of K and λ are moderate, such as $K = 4$ and $\lambda = 0.1$, activation intensity of the DA module is moderate, too. The precision and recall are both high, because the similarity module can effectively improve the feature distribution. It further improves the response intensity of similar features and suppresses the background noise.

In conclusion, the combination of the DA module and the similarity module is very efficient but requires a good control of the activation intensity of the DA module. If the activation intensity is too low, the performance will not be improved. If the activation intensity is too high, there will be many false alarms. When the activation intensity of the DA module is moderate, it can effectively solve the problem of object loss and improve the precision and recall rate of object detection.

6. Conclusion and future work

In this paper, we proposed the *Hierarchical Fusion Based Remote Sensing Object Location* method (HF-RSOL) and the *Similarity-Divergent Activation Based Remote Sensing Object Detection* method (SDA-RSOD) building upon it. The HF-RSOL utilizes the shallow feature maps to locate objects and uses the deep feature map to filter the target response areas. Fusing these maps solves the problem of inaccurate object location and allows for an efficient application of weakly supervised learning for object detection from RS images.

The SDA-RSOD adds the DA module to the third layer of the basic network ResNet34 of the HF-RSOL to improve the target response strength of the low-response areas. To account for the characteristic dense distributions of objects in RS images, a similarity module is introduced to further improve the features in the shallow feature map. It adjusts the response intensity of similar objects and suppresses background noise to ensure that the object features are not lost when they are passed to the next layer. The novel combination of both modules significantly improves the network performance.

The effectiveness of our methods was rigorously verified with comprehensive experiments, which we also provide in the repository [67]. We found that the information fusion step in HF-RSOL is very effective for the location of objects on RS images and that the SDA-RSOD is the overall best method for RS object detection in our study. It can outperform the state-of-the-art CAM works used for comparison, all of which have been published within the past three years. On the WSADD, for example, it detects almost 8.5 times as many objects while reducing the false positives by 50%. Our approach significantly extends of what CAM-based methods can achieve. It is also highly competitive to and more robust than the state-of-the-art MIL approaches.

In the future, we will improve and generalize our approach. We will aim to combine our SDA-RSOD, which can already improve the performance of the detection of small but similar objects, with other concepts to detect very small objects such as vehicles [33]. One important line of future work concerns the currently available backbones, such as ResNet34, which are often based on CNNs designed for natural scenes. They cannot adapt optimally to the characteristics of RS data and there may be more suitable models for a specific RS task. Therefore, in the next step of our work, we will explore different network models to obtain better RS target response information and to better learn representative features from the feature maps. As third element of our future work, we will also aim to incorporate a more advanced handling of clouds [35]. Fourth, we will improve the detection performance for very large objects such as airports by making use of the recently emerging part-based approaches [34]. Finally, we intend to study the fusion between the multi-instance learning model and the SDA-RSOD to utilize their complementary advantages for WSOD.

CRediT authorship contribution statement

Zhi-Ze Wu: Conceptualization, Methodology, Writing – original draft. **Jian Xu:** Software. **Yan Wang:** Visualization. **Fei Sun:** Funding acquisition. **Ming Tan:** Conceptualization. **Thomas Weise:** Supervision, Writing – review & editing.

Acknowledgments

We acknowledge support from the Youth Project of the Provincial Natural Science Foundation of Anhui, China 1908085QF285, the Talent Fund of Hefei University, China 18-19RC34, the University Natural Sciences Research Project of Anhui Province, China KJ2020A0661, the National Natural Science Foundation of China under grant 61673359, the Key Research Plans of Anhui, China 201904d07020002 and 202104d07020006, as well as the Hefei Specially Recruited Foreign Expert program.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] G. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, X. Lu, AID: a benchmark data set for performance evaluation of aerial scene classification, IEEE Trans. Geosci. Remote Sens. 55 (7) (2017) 3965–3981, http://dx.doi.org/10.1109/TGRS.2017.2685945.

[2] W. Zhou, S. Newsam, C. Li, Z. Shao, Patternnet: A benchmark dataset for performance evaluation of remote sensing image retrieval, ISPRS J. Photogr. Remote Sens. 145 (2018) 197–209, http://dx.doi.org/10.1016/j.isprsjprs.2018.01.004.

[3] Z.-Z. Wu, S.-H. Wan, X.-F. Wang, M. Tan, L. Zou, X.-L. Li, Y. Chen, A benchmark data set for aircraft type recognition from remote sensing images, Appl. Soft Comput. 89 (2020) 106132, http://dx.doi.org/10.1016/j.asoc.2020.106132.

[4] Y. Li, J. Ma, Y. Zhang, Image retrieval from remote sensing big data: A survey, Inf. Fusion 67 (2021) 94–115, http://dx.doi.org/10.1016/j.inffus.2020.10.008.

[5] Z. Zheng, Y. Zhong, A. Ma, X. Han, J. Zhao, Y. Liu, L. Zhang, HyNet: hyper-scale object detection network framework for multiple spatial resolution remote sensing imagery, ISPRS J. Photogr. Remote Sens. 166 (2020) 1–14, http://dx.doi.org/10.1016/j.isprsjprs.2020.04.019.

[6] Y. Du, W. Song, Q. He, D. Huang, A. Liotta, C. Su, Deep learning with multi-scale feature fusion in remote sensing for automatic oceanic eddy detection, Inf. Fusion 49 (2019) 89–99, http://dx.doi.org/10.1016/j.inffus.2018.09.006.

[7] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444, http://dx.doi.org/10.1038/nature14539.

[8] H. Zhu, M. Ma, W. Ma, L. Jiao, S. Hong, J. Shen, B. Hou, A spatial-channel progressive fusion resnet for remote sensing classification, Inf. Fusion (2020) http://dx.doi.org/10.1016/j.inffus.2020.12.008, Early access.

[9] K. Li, G. Wan, G. Cheng, L. Meng, J. Han, Object detection in optical remote sensing images: A survey and a new benchmark, ISPRS J. Photogr. Remote Sens. 159 (2020) 296–307, http://dx.doi.org/10.1016/j.isprsjprs.2019.11.023.

[10] P. Ding, Y. Zhang, W.-J. Deng, P. Jia, A. Kuijper, A light and faster regional convolutional neural network for object detection in optical remote sensing images, ISPRS J. Photogr. Remote Sens. 141 (2018) 208–218, http://dx.doi.org/10.1016/j.isprsjprs.2018.05.005.

[11] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Y. Bengio, Y. LeCun (Eds.), Conference Track Proceedings of the 3rd International Conference on Learning Representations (ICLR'15), (2015) 7-9, San Diego, CA, USA, 2014, 1–14. http://arxiv.org/abs/1409.1556.

[12] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Jun. (2016) 27-30, IEEE Computer Society, Las Vegas, NV, USA, 2016, pp. 770–778, http://dx.doi.org/10.1109/CVPR.2016.90.

[13] K. Fu, W. Dai, Y. Zhang, Z. Wang, M. Yan, X. Sun, MultiCAM: multiple class activation mapping for aircraft recognition in remote sensing images, Remote Sens. 11 (5) (2019) 544, http://dx.doi.org/10.3390/rs11050544.

[14] S. Ren, K. He, R.B. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, IEEE Trans. Pattern Anal. Mach. Intell. 39 (6) (2017) 1137–1149, http://dx.doi.org/10.1109/TPAMI.2016.2577031.

[15] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Jun. (2016) 27-30, IEEE Computer Society, Las Vegas, NV, USA, 2016, pp. 2921–2929, http://dx.doi.org/10.1109/CVPR.2016.319.

[16] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, S. Yan, Object region mining with adversarial erasing: a simple classification to semantic segmentation approach, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17), Jul. (2017) 21-26, IEEE Computer Society, Honolulu, HI, USA, 2017, pp. 6488–6496, http://dx.doi.org/10.1109/CVPR.2017.687.

[17] X. Zhang, Y. Wei, J. Feng, Y. Yang, T.S. Huang, Adversarial complementary learning for weakly supervised object localization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18), Jun. (2018) 18-22, IEEE Computer Society, Salt Lake City, UT, USA, 2018, pp. 1325–1334, http://dx.doi.org/10.1109/CVPR.2018.00144.

[18] D. Kim, D. Cho, D. Yoo, I. So Kweon, Two-phase learning for weakly supervised object localization, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV'17), Oct. (2017) 22-29, IEEE Computer Society, Venice, Italy, 2017, pp. 3554–3563, http://dx.doi.org/10.1109/ICCV.2017.382.

[19] X. Zhang, Y. Wei, G. Kang, Y. Yang, T.S. Huang, Self-produced guidance for weakly-supervised object localization, in: V. Ferrari, M. Hebert, C. Sminchisescu, Y. Weiss (Eds.), Proceedings of the 15th European Conference on Computer Vision (ECCV'18), Sep. 8-14, Munich, Germany, Part XII, in: Lecture Notes in Computer Science (LNCS), vol. 11216, Springer, 2018, pp. 610–625, http://dx.doi.org/10.1007/978-3-030-01258-8_37.

[20] S. Yun, D. Han, S. Chun, S.J. Oh, Y. Yoo, J. Choe, CutMix: Regularization strategy to train strong classifiers with localizable features, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV'19), Oct. 27-Nov. 2, 2019, Seoul, Korea, IEEE, 2019, pp. 6022–6031, http://dx.doi.org/10.1109/ICCV.2019.00612.

[21] J. Choe, S.J. Oh, S. Lee, S. Chun, Z. Akata, H. Shim, Evaluating weakly supervised object localization methods right, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20), Jun. (2020) 13-19, IEEE, Seattle, WA, USA, 2020, pp. 3130–3139, http://dx.doi.org/10.1109/CVPR42600.2020.00320.

[22] R. Qiao, A. Ghodsi, H. Wu, Y. Chang, C. Wang, Simple weakly supervised deep learning pipeline for detecting individual red-attacked trees in VHR remote sensing images, Remote Sens. Lett. 11 (7) (2020) 650–658, http://dx.doi.org/10.1080/2150704X.2020.1752410.

[23] Z.-Z. Wu, T. Weise, Y. Wang, Y. Wang, Convolutional neural network based weakly supervised learning for aircraft detection from remote sensing image, IEEE Access 8 (2020) 158097–158106, http://dx.doi.org/10.1109/ACCESS.2020.3019956.

[24] Y. Li, W. Chen, Y. Zhang, C. Tao, R. Xiao, Y. Tan, Accurate cloud detection in high-resolution remote sensing imagery by weakly supervised deep learning, Remote Sens. Environ. 250 (2020) 112045, http://dx.doi.org/10.1016/j.rse.2020.112045.

[25] Y. Li, Y. Zhang, X. Huang, A.L. Yuille, Deep networks under scene-level supervision for multi-class geospatial object detection from remote sensing images, ISPRS J. Photogr. Remote Sens. 146 (2018) 182–196, http://dx.doi.org/10.1016/j.isprsjprs.2018.09.014.

[26] Y. Chen, Y. Cao, H. Hu, L. Wang, Memory enhanced global-local aggregation for video object detection, in: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20), Jun. (2020) 13-19, IEEE, Seattle, WA, USA, 2020, pp. 10334–10343, http://dx.doi.org/10.1109/CVPR42600.2020.01035.

[27] L. Chen, A. Hermans, G. Papandreou, F. Schroff, P. Wang, H. Adam, MaskLab: Instance segmentation by refining object detection with semantic and direction features, in: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18), Jun. (2018) 18-22, IEEE Computer Society, Salt Lake City, UT, USA, 2018, pp. 4013–4022, http://dx.doi.org/10.1109/CVPR.2018.00422.

[28] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Gläser, F. Timm, W. Wiesbeck, K. Dietmayer, Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges, IEEE Trans. Intell. Transpor. Syst. 22 (3) (2021) 1341–1360, http://dx.doi.org/10.1109/TITS.2020.2972974.

[29] M. Simon, K. Amende, A. Kraus, J. Honer, T. Sämann, H. Kaulbersch, S. Milz, H. Gross, Complexer-YOLO: Real-time 3D object detection and tracking on semantic point clouds, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops'19), Jun. (2019) 16-20, Computer Vision Foundation / IEEE, Long Beach, CA, USA, 2019, pp. 1190–1199, http://dx.doi.org/10.1109/CVPRW.2019.00158.

[30] L. Zhang, J. Zhang, Z. Lin, H. Lu, Y. He, CapSal: Leveraging captioning to boost semantics for salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'19), Jun. (2019) 16-20, Computer Vision Foundation / IEEE, Long Beach, CA, USA, 2019, pp. 6024–6033, http://dx.doi.org/10.1109/CVPR.2019.00618.

[31] D. Zhang, D. Meng, J. Han, Co-saliency detection via a self-paced multiple-instance learning framework, IEEE Trans. Pattern Anal. Mach. Intell. 39 (5) (2017) 865–878, http://dx.doi.org/10.1109/TPAMI.2016.2567393.

[32] Z.-Z. Wu, Weakly supervised airplane detection dataset: WSADD, 2020, http://dx.doi.org/10.5281/zenodo.3843229.

[33] P. Gao, T. Tian, L. Li, J. Ma, J. Tian, DE-CycleGAN: An object enhancement network for weak vehicle detection in satellite images, IEEE J. Sel. Top. Appl. Earth Obser. Remote Sens. 14 (2021) 3403–3414, http://dx.doi.org/10.1109/JSTARS.2021.3062057.

[34] X. Sun, P. Wang, C. Wang, Y. Liu, K. Fu, PBNet: Part-based convolutional neural network for complex composite object detection in remote sensing imagery, ISPRS J. Photogr. Remote Sens. 173 (2021) 50–65, http://dx.doi.org/10.1016/j.isprsjprs.2020.12.015.

[35] Q. He, X. Sun, Z. Yan, K. Fu, DABNet: Deformable contextual and boundary-weighted network for cloud detection in remote sensing images, IEEE Trans. Geosci. Remote Sens. (2021) 1–16, http://dx.doi.org/10.1109/TGRS.2020.3045474, Early access.

[36] G. Cheng, J. Yang, D. Gao, L. Guo, J. Han, High-quality proposals for weakly supervised object detection, IEEE Trans. Image Process. 29 (2020) 5794–5804, http://dx.doi.org/10.1109/TIP.2020.2987161.

[37] H. Bilen, A. Vedaldi, Weakly supervised deep detection networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Jun. (2016) 27-30, IEEE Computer Society, Las Vegas, NV, USA, 2016, pp. 2846–2854, http://dx.doi.org/10.1109/CVPR.2016.311.

[38] V. Kantorov, M. Oquab, M. Cho, I. Laptev, ContextLocNet: context-aware deep network models for weakly supervised localization, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Proceedings of the 14th European Conference on Computer Vision (ECCV'16), Oct.(2016) 11-14, in: Lecture Notes in Computer Science (LNCS), vol. 9909, Springer, Amsterdam, The Netherlands, Part V, 2016, pp. 350–365, http://dx.doi.org/10.1007/978-3-319-46454-1_22.

[39] P. Tang, X. Wang, X. Bai, W. Liu, Multiple instance detection network with online instance classifier refinement, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17), Jul. (2017) 21-26, IEEE Computer Society, Honolulu, HI, USA, 2017, pp. 3059–3067, http://dx.doi.org/10.1109/CVPR.2017.326.

[40] Y. Zhang, Y. Bai, M. Ding, Y. Li, B. Ghanem, W2F: a weakly-supervised to fully-supervised framework for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18), Jun. (2018) 18-22, IEEE Computer Society, Salt Lake City, UT, USA, 2018, pp. 928–936, http://dx.doi.org/10.1109/CVPR.2018.00103.

[41] F. Wan, C. Liu, W. Ke, X. Ji, J. Jiao, Q. Ye, C-MIL: continuation multiple instance learning for weakly supervised object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'19), Jun. (2019) 16-20, Computer Vision Foundation / IEEE, Long Beach, CA, USA, 2019, pp. 2199–2208, http://dx.doi.org/10.1109/CVPR.2019.00230.

[42] A. Arun, C.V. Jawahar, M.P. Kumar, Dissimilarity coefficient based weakly supervised object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'19), Jun. (2019) 16-20, Computer Vision Foundation / IEEE, Long Beach, CA, USA, 2019, pp. 9432–9441, http://dx.doi.org/10.1109/CVPR.2019.00966.

[43] K. Yang, D. Li, Y. Dou, Towards precise end-to-end weakly supervised object detection network, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV'19), Oct. 27 - Nov. 2, IEEE, Seoul, Korea, 2019, pp. 8371–8380, http://dx.doi.org/10.1109/ICCV.2019.00846.

[44] Y. Shen, R. Ji, Y. Wang, Y. Wu, L. Cao, Cyclic guidance for weakly supervised joint detection and segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'19), Jun. (2019) 16-20, Computer Vision Foundation / IEEE, Long Beach, CA, USA, 2019, pp. 697–707, http://dx.doi.org/10.1109/CVPR.2019.00079.

[45] X. Feng, J. Han, X. Yao, G. Cheng, Progressive contextual instance refinement for weakly supervised object detection in remote sensing images, IEEE Trans. Geosci. Remote Sens. 58 (11) (2020) 8002–8012, http://dx.doi.org/10.1109/TGRS.2020.2985989.

[46] X. Yao, X. Feng, J. Han, G. Cheng, L. Guo, Automatic weakly supervised object detection from high spatial resolution remote sensing images via dynamic curriculum learning, IEEE Trans. Geosci. Remote Sens. 59 (2021) 675–685, http://dx.doi.org/10.1109/TGRS.2020.2991407.

[47] X. Feng, J. Han, X. Yao, G. Cheng, TCANet: triple context-aware network for weakly supervised object detection in remote sensing images, IEEE Trans. Geosci. Remote Sens. (2020) 1–10, http://dx.doi.org/10.1109/TGRS.2020.3030990, Early access.

[48] H. Xue, C. Liu, F. Wan, J. Jiao, X. Ji, Q. Ye, DANet: Divergent activation for weakly supervised object localization, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV'19), Oct. 27-Nov, Vol. 2, IEEE, Seoul, Korea, 2019, pp. 6588–6597, http://dx.doi.org/10.1109/ICCV.2019.00669.

[49] M. Rey-Area, E. Guirado, S. Tabik, J. Ruiz-Hidalgo, FuCiTNet: Improving the generalization of deep learning networks by the fusion of learned class-inherent transformations, Inf. Fusion 63 (2020) 188–195, http://dx.doi.org/10.1016/j.inffus.2020.06.015.

[50] Y. Wei, H. Xiao, H. Shi, Z. Jie, J. Feng, T.S. Huang, Revisiting dilated convolution: A simple approach for weakly- and semi-supervised semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18), Jun. (2018) 18-22, IEEE Computer Society, Salt Lake City, UT, USA, 2018, pp. 7268–7277, http://dx.doi.org/10.1109/CVPR.2018.00759.

[51] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: P. L. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 25: The 26th Annual Conference on Neural Information Processing Systems (NIPS'21). Proceedings of a meeting held Dec. (2012) 3-6, Lake Tahoe, NV, USA, 2012, pp. 1106–1114. https://proceedings.neurips.cc/paper/2012.

[52] J. Redmon, S.K. Divvala, R.B. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Jun. (2016) 27-30, IEEE Computer Society, Las Vegas, NV, USA, 2016, pp. 779–788, http://dx.doi.org/10.1109/CVPR.2016.91.

[53] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'19), Jun. (2019) 16-20, Computer Vision Foundation / IEEE, Long Beach, CA, USA, 2019, pp. 3146–3154, http://dx.doi.org/10.1109/CVPR.2019.00326.

[54] J. Fu, J. Liu, J. Jiang, Y. Li, Y. Bao, H. Lu, Scene segmentation with dual relation-aware attention network, IEEE Trans. Neural Netw. Learn. Syst. (2020) 1–14, http://dx.doi.org/10.1109/TNNLS.2020.3006524, Early Access.

[55] K.R. Sloan, Analysis of dot product space shape descriptions, IEEE Trans. Pattern Anal. Mach. Intell. 4 (1) (1982) 87–90, http://dx.doi.org/10.1109/TPAMI.1982.4767202.

[56] S. Jin, H. Yao, X. Sun, S. Zhou, L. Zhang, X. Hua, Deep saliency hashing for fine-grained retrieval, IEEE Trans. Image Process. 29 (2020) 5336–5351, http://dx.doi.org/10.1109/TIP.2020.2971105.

[57] T. Do, T. Hoang, D.L. Tan, H. Le, T.V. Nguyen, N. Cheung, From selective deep convolutional features to compact binary representations for image retrieval, ACM Trans. Multimed. Comput. Commun. Appl. 15 (2) (2019) 43:1–43:22, http://dx.doi.org/10.1145/3314051.

[58] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Jun. (2016) 27-30, IEEE Computer Society, Las Vegas, NV, USA, 2016, pp. 770–778, http://dx.doi.org/10.1109/CVPR.2016.90.

[59] J. Deng, W. Dong, R. Socher, L. Li, K. Li, F. Li, ImageNet: A large-scale hierarchical image database, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'09), Jun. (2009) 20-25, IEEE Computer Society, Miami, FL, USA, 2009, pp. 248–255, http://dx.doi.org/10.1109/CVPR.2009.5206848.

[60] G. Cheng, J. Han, P. Zhou, L. Guo, Multi-class geospatial object detection and geographic image classification based on collection of part detectors, ISPRS J. Photogr. Remote Sens. 98 (2014) 119–132, http://dx.doi.org/10.1016/j.isprsjprs.2014.10.002.

[61] G. Cheng, P. Zhou, J. Han, Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images, IEEE Trans. Geosci. Remote Sens. 54 (12) (2016) 7405–7415, http://dx.doi.org/10.1109/TGRS.2016.2601622.

[62] G. Cheng, J. Han, A survey on object detection in optical remote sensing images, ISPRS J. Photogr. Remote Sens. 117 (2016) 11–28, http://dx.doi.org/10.1016/j.isprsjprs.2016.03.014.

[63] Z. Lu, K. Deb, V.N. Boddeti, MUXConv: Information multiplexing in convolutional neural networks, in: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20), Jun. (2020) 13-19, IEEE, Seattle, WA, USA, 2020, pp. 12041–12050, http://dx.doi.org/10.1109/CVPR42600.2020.01206.

[64] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Y. Bengio, Y. LeCun (Eds.), Conference Track Proceedings of the 3rd International Conference on Learning Representations (ICLR'15), (2015) 7-9, San Diego, CA, USA, 2015, 2–14. http://arxiv.org/abs/1409.1556.

[65] S.M. Beitzel, E.C. Jensen, O. Frieder, Gmap, in: M. T. Özsu L. Liu (Ed.), Encyclopedia of Database Systems, Springer US, Boston, MA, 2009, p. 1256, http://dx.doi.org/10.1007/978-0-387-39940-9_493.

[66] E.M. Voorhees, L.P. Buckland (Eds.), Appendix: Common evaluation measures, in: Proceedings of the Fifteenth Text REtrieval Conference (TREC 2006). Nov. (2006) 14–17, SP 500-272 of NIST Special Publication, National Institute of Standards and Technology (NIST), Gaithersburg, ML, USA, 2006, https://trec.nist.gov/pubs/trec15/appendices/CE.MEASURES06.pdf.

[67] Z. Wu, T. Weise, Experimental data for the paper 'Hierarchical fusion and divergent activation based weakly supervised learning for object detection from remote sensing images', 2021, http://dx.doi.org/10.5281/zenodo.4420286.