Weakly Supervised Learning for Target Detection in Remote Sensing Images

Dingwen Zhang, Junwei Han, Gong Cheng, Zhenbao Liu, Shuhui Bu, and Lei Guo

***Abstract*—In this letter, we develop a novel framework of leveraging weakly supervised learning techniques to efficiently detect targets from remote sensing images, which enables us to reduce the tedious manual annotation for collecting training data while maintaining the detection accuracy to large extent. The proposed framework consists of a weakly supervised training procedure to yield the detectors and an effective scheme to detect targets from testing images. Comprehensive evaluations on three benchmarks which have different spatial resolutions and contain different types of targets as well as the comparisons with traditional supervised learning schemes demonstrate the efficiency and effectiveness of the proposed framework.**

***Index Terms*—Remote sensing image (RSI), target detection, weakly supervised learning (WSL).**

I. Introduction

NOWADAYS, detection of valuable targets from remote sensing images (RSIs) has become one of the most fundamental and challenging tasks. Some researchers applied unsupervised models without training procedure to perform target detection. For example, Tello *et al.* [1] proposed to detect ships in synthetic aperture radar images using the discrete wavelet transform. Sirmacek [2] detected urban areas and buildings using the SIFT keypoints and graph theory. However, these methods may be only effective for detecting targets with simple appearance and small variations in less complex background.

Most approaches built supervised learning models for target detection by taking advantage of the prior information obtained from training samples. Specifically, the work of [3] trained the SVM classifier based on the extracted feature to recognize targets. The work in [4] applied the saliency model and discriminative sparse coding for multiclass geospatial target detection. These approaches can achieve a good accuracy only when the training examples are manually labeled using bounding boxes. However, manual annotation is generally expensive and time-consuming or sometimes even unreliable. It may become more difficult for labeling data in RSIs due to the reason that RSIs always contain complex textures and the size of the RSI is generally very big, whereas the coverage of the target is relatively small. Subjects have a difficulty in focusing their attention on such small regions for detailed annotation. Moreover, the artificial annotations may tend to be less accurate and unreliable when the targets are occluded or camouflaged.

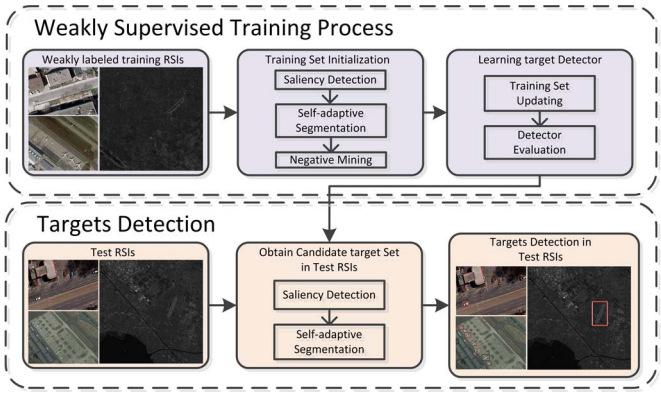

Fig. 1. Flowchart of the proposed framework.

One way to alleviate the work of human annotation is to use the semisupervised learning models. Such models [5], [6] apply active learning and kernel methods to automatically pick the most informative unlabeled samples and then explore the information from both the labeled examples and the unlabeled examples for the target detection. Although the semisupervised learning model can decrease human's labor to a certain extent, it still requires a considerable number of positive examples manually localized by subjects [7].

This letter attempts to develop a novel framework by using weakly supervised learning (WSL) techniques [8], [9] for the purpose of further reducing or minimizing the human's labor for collecting training data while not affecting the accuracy of the detector significantly. In contrast to supervised learning, WSL just requires subjects to annotate each image in the training set with a weak label which only indicates whether the image contains certain targets or not, whereas the locations and sizes of the targets are not necessary. As one of the earliest efforts of using WSL for target detection from RSIs, the proposed framework shown in Fig. 1 consists of two components: 1) a novel weakly supervised training process to train target detectors based on the training images with weak labels, which adopts saliency-based self-adaptive segmentation and negative mining approach to initialize the training samples and an iteratively training scheme to refine the detector and 2) an efficient target detection approach which applies the candidate target selection process to improve the efficiency of detection using the trained detector.

II. Proposed Framework

Given a number of weakly labeled RSIs, which do not contain the location and size information about the targets,

Manuscript received December 2, 2013; revised March 31, 2014, July 15, 2014, and August 4, 2014; accepted September 4, 2014. Date of publication October 3, 2014; date of current version October 31, 2014. This work was supported by the National Natural Science Foundation of China under Grants 91120005, 61473231, 61202185, and NPU-Z2013105. *(Corresponding author: Junwei Han.)*

D. Zhang, J. Han, G. Cheng, and L. Guo are with the School of Automation, Northwestern Polytechnical University, Xi'an 710072, China (e-mail: junweihan2010@gmail.com).

Z. Liu and S. Bu are with the School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, China.

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

Digital Object Identifier 10.1109/LGRS.2014.2358994

the proposed WSL training scheme obtains the initial training examples via saliency-based self-adaptive segmentation and negative mining first. Then, an iterative training process is designed to refine the training examples as well as the target detector gradually. Based on the trained target detector, a candidate-patch-based scheme is then adopted to detect targets effectively and efficiently.

A. Negative Training Set Initialization

In weakly labeled training RSIs, negative training examples are very easy to obtain because the negative RSIs definitely do not contain any target. Therefore, we randomly select a number of patches in negative RSIs to form the negative example set $X^- = \{x_j^-\},\ j = 1, 2, \ldots, m.\ x_j^-$ is one of the m negative examples which is described by a set of features independent from the size of the patch. By considering the fact that the imbalanced training data set may reduce the performance of the classifier, the initial negative training set X_0^- is established by randomly selecting negative examples in X^- with the same number of examples in the initial positive training set X_0^+, which will be described in the next section.

B. Positive Training Set Initialization

Unlike the negative training set, the positive training set is extremely hard to generate because, in weakly labeled positive RSIs, there is no concrete information about the location, shape, and size of the targets. In this letter, we adopt saliency-based self-adaptive segmentation and negative mining approach to initialize the positive training.

Saliency-Based Self-Adaptive Segmentation: The first observation is that targets generally are distinct from the background in an RSI. Many previous works have demonstrated that computational saliency models can effectively characterize the distinctiveness between the salient region and the background region. For the RSIs in the positive training RSIs, the low- and midlevel features described in [10] are extracted for every pixel of the image. The low-level features include local contrast of intensity, orientation, and global color contrast. The midlevel features include the saliency maps obtained by four previous saliency models [11]–[14]. After being normalized to [0, 1], all of these features are linearly combined (with equivalent weights) to yield the overall saliency map, which measures the distinctiveness of each pixel in the image. Based on the saliency map, a self-adaptive segmentation can be performed to obtain positive examples in each positive RSI by

$$\text{thresh} = \frac{t}{W \times H} \times \sum_{x=0}^{W-1} \sum_{y=0}^{H-1} S(x, y) \tag{1}$$

where W and H are the width and height of the original image, $S(x, y) \in [0, 1]$ is the saliency value of the pixel at position (x, y), and t is a parameter for generating the appropriate bounding boxes for the positive examples. In this letter, multiple thresholds denoted by $T = \{t_1, t_2, \ldots, t_p\}$ are applied to segment the saliency maps to obtain the initial positive examples in order to deal with the variation of the target size and the resolution of the RSI. After segmentation, pixels that are distinctive from the background are segmented, and connected pixels which may form an object are indicated by a bounding box. The bounding box is then refined by finding the area enclosing 99.9% of its edge energy as suggested in [15]. Afterward, the patches labeled by the bounding boxes are regarded as the positive examples x_i^+. A few examples are shown in the third column of Fig. 2, where the bounding boxes in red color indicate the positive examples. Finally, the formed positive example set $X^+ = \{x_i^+\},\ i = 1, 2, \ldots, n$ can contain targets with different orientations, shapes, and scales. It also contains a number of false positives which can be removed by the negative mining described in the next section.

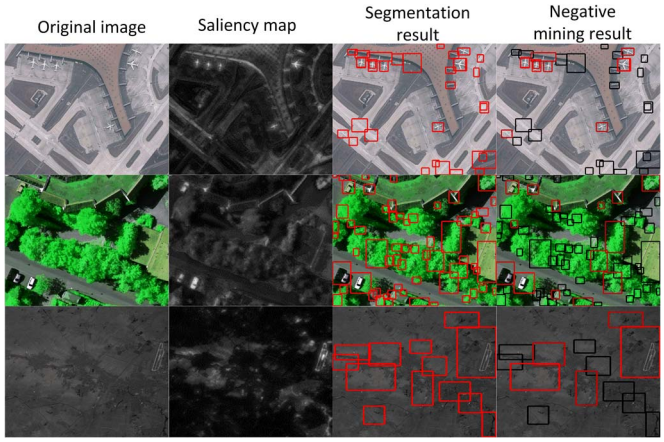

Fig. 2. Some examples for saliency map, segmentation, and negative mining in positive training set initialization.

Negative Mining: The second observation is that the appearance of the targets is normally different from that of the negative examples in negative RSIs. We use the negative mining [8] to select patches from the positive example set X^+ to form the initial positive training set X_0^+. Specifically, it has been declared that each of the positive RSIs contains at least one target, whereas every negative RSI does not contain any target. Given an example patch $x_i^+ \in X^+$ (or $x_j^- \in X^-$), it is represented by a feature vector $\mathbf{f}_i^+$ (or $\mathbf{f}_j^-$). Then, we score every positive example in the positive example set X^+ and select some highest scored patches to form the initial positive training set X_0^+. The negative mining algorithm accomplishes this by selecting the positive examples largely different with their most similar examples in the negative example set X^- by

$$X_0^+ = \left\{x_i^+ | D\left(x_i^+\right) > \sigma,\ x_i^+ \in X^+\right\} \tag{2}$$

$$D\left(x_i^+\right) = \min_{j \in [1,m]} \left\|\mathbf{f}_i^+ - \mathbf{f}_j^-\right\|_1 \tag{3}$$

where $\|\cdot\|_1$ is the L_1 norm and σ is a parameter set empirically to balance the precision and recall. The normalized $D(x_i^+)$ indicates the similarity between a positive example and all of the examples in the negative example set. Here, a fast nearest neighbor look-up algorithm named KD-tree-based approximate nearest neighbor algorithm is adopted to handle the large volume of data efficiently. In the fourth column of Fig. 2, some negative mining results are shown, where the true positives indicated by the red rectangles are selected to form X_0^+, whereas the false positives indicated by the black rectangles are removed by negative mining.

C. Iterative Detector Training

As shown in Fig. 3, the target detector training process includes a training set updating step and a detector evaluation

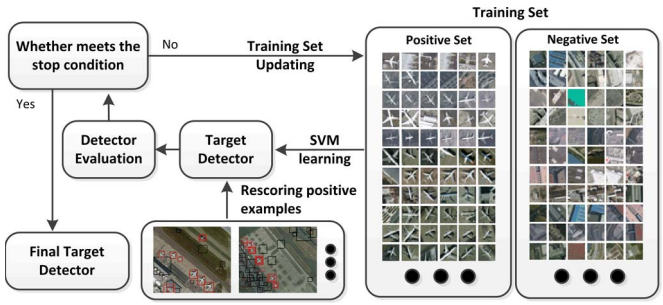

Fig. 3. Framework of iterative detector training.

step. These two steps are iteratively performed until the best performance is reached.

Training Set Updating: The aim of this section is to refine the training set iteratively and optimize the target detector simultaneously. First, we utilize the initial training set $X_0^+ \cup X_0^-$ to train a classifier. As suggested in [3] and [4], the linear SVM is adopted as

$$\min_{w_1,b_1} \frac{1}{2}\|w_1\|^2 \quad \text{s.t. } y\left[\left(w_1^T\mathbf{f}+b_1\right)\right]-1\geq 0 \tag{4}$$

where $\mathbf{f}$ is the feature vector of training example $x \in X_0^+ \cup X_0^-$ and $y \in \{+1,-1\}$ is its label. Afterward, the classifier is used to update the positive training set by

$$\begin{aligned}\text{Score}\left(x_i^+\right) &= w_1^T\mathbf{f}_i^+ + b_1\\ X_1^+ &= \left\{x_i^+ | \text{Score}\left(x_i^+\right) > \sigma,\ x_i^+ \in X^+\right\}\end{aligned} \tag{5}$$

where X_1^+ is the updated positive training set in the first iteration. Next, the same number of negative examples randomly selected from X^- is used to generate the new negative training set X_1^-. As shown in Fig. 3, this process can run iteratively to jointly update the training examples as well as improve the target detector gradually.

Detector Evaluation: We propose to use a negative evaluation mechanism to evaluate the target detector trained in each iteration because only the negative training set can be accessed precisely in the WSL scheme and the minimization of false positive results is a key component for detector evaluation. Specifically, for a target classifier (w,b), we use it to classify each negative patch in negative example set X^- and calculate the false rate by using $\text{FR} = |X_{\text{false}}^-|/|X^-|$, where $X_{\text{false}}^- = \{x_j^- | w^T\mathbf{f}_j^- + b > 0\}$ and $|\cdot|$ refers to the number of elements. Normally, the false rate decreases in the first several iterations continually and then begins to increase, which means that the performance of the target detector is improved gradually and then becomes worse. Therefore, we stop the iteration when the local minimal false rate is achieved, and the target detector trained in this iteration is selected as the final target detector.

D. Target Detection

To detect the targets in the RSIs, many conventional methods use a sliding window to scan over the entire image. However, this exhaustive search scheme is an extremely time-consuming work and cannot obtain the optimal detection result. To solve this problem, a candidate-patch-based target detection scheme is adopted in our framework. For a given testing RSI, the proposed saliency-based self-adaptive segmentation (described in Section II-B) is applied to obtain the candidate patches. Afterward, the target detector trained by the proposed WSL-based framework is used to classify these patches into targets or background.

TABLE I
INFORMATION ABOUT THE THREE EVALUATION DATABASES

Data Set	Dimension (pixels)	Spatial Resolution	Target Area (pixels)
Google Earth	about 1000×800	About 0.5m	700~25488
ISPRS	about 900×700	8-15cm	1150~11976
Landsat	400×400	30m	1760~15570

III. EXPERIMENTAL RESULTS

A. Data Set Description and Experimental Setup

We evaluated the proposed work using three different RSI benchmark databases, which have different spatial resolutions and contain different targets. The details are shown in Table I. The Google Earth data set contains 120 high-resolution RSIs of a large number of different airplanes. The ISPRS data set is a very high resolution RSI data set which contains 100 images of vehicles provided by the German Association of Photogrammetry and Remote Sensing [16]. The Landsat data set is acquired by Landsat-7 ETM+ sensor and includes 180 infrared RSIs of a variety of airports in China. Fig. 4 shows a number of image samples. As can be seen, the targets in the different data sets have different sizes, orientations, and colors.

In the experiments, we applied the proposed framework to detect airplanes, vehicles, and airports from Google Earth data set, ISPRS data set, and Landsat data set, respectively. Specifically, we randomly selected 70 RSIs in the Google Earth data set for training, and the remaining 50 RSIs were used for testing. In the ISPRS data set, the training set includes 60 RSIs, and the testing set includes 40 RSIs. In the Landsat data set, the training set and testing set contain 123 and 57 RSIs, respectively. In the experiments, we empirically set the parameters in our framework as $T = \{1.5, 1.8, 2\}$ for multithreshold segmentation and $\sigma = 0.85$ in (2) and (5), and adopt the bag-of-visual-words (BOW) feature [17], the locality constrained linear coding (LLC) [18] feature, and the pyramid histograms of oriented gradients (pHOG) [19] feature to represent each of the training example. Specially, BOW characterizes each patch via a histogram of visual words from a codebook. LLC uses locality constraint to select several (five in this letter) similar bases from the codebook and learns a linear combination weight of these bases to reconstruct each descriptor. The pHOG feature represents the shape property of the image patches by histograms of orientation gradients which are discretized into 16 bins with orientations in the range [0, 180]. In our implementation, the codebook used in BOW and LLC was generated by extracting 128-dimensional SIFT descriptors [17] in training images and then clustering them to 1024 visual words via k-means algorithm [17]. For better dealing with the target variations in rotation, we used the global level of the pyramid representation in pHOG and LLC.

B. Evaluation of the Training Iteration

To demonstrate the effectiveness of the proposed detector training and evaluation method, we show the detection performance (based on the pHOG feature) of each iteration over three data sets in Fig. 5. Specifically, the false rate curves in Fig. 5(a) indicate the variation of false detection results in the

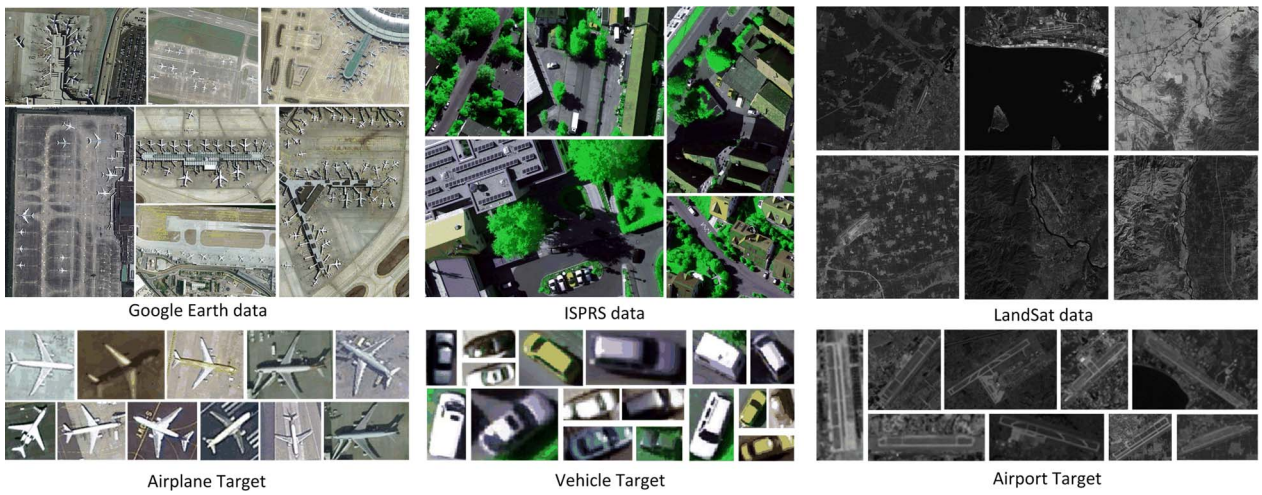

Fig. 4. Some samples from three benchmark data sets.

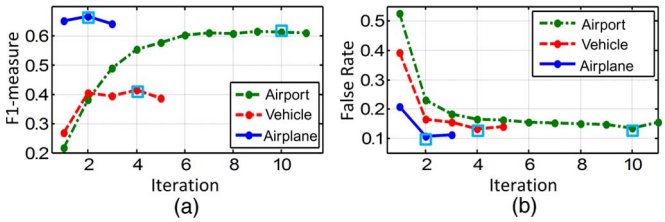

Fig. 5. Detector performance evaluation in iterative detector training.

negative training set. From these curves, we can see that the detection performance increases in the first several iterations and then begins to decrease. The iteration is terminated at the minimum point (marked on the curve) to generate the final target detector. Fig. 5(b) shows the F1-measure curves which indicate the change of target detector's performance during the iteration. The marked points on these curves correspond to the minimum location in the false rate curves and the localization performance of the final detectors. The experimental results can demonstrate that the performance of the detector can be improved with the iterations and the detector evaluation scheme is effective for terminating the iterations.

C. Evaluation of Target Detection

Fig. 6 shows a number of detection results by using the proposed framework. In Fig. 6, red, black, and yellow rectangles indicate the true-positive, false-positive, and miss alarm results, respectively. As can be seen, the proposed WSL-based target detector can effectively detect targets from different data sets with different spatial resolutions and is robust to the scale, rotation, and shape variation of the target.

To evaluate the experimental results quantitatively, we adopted the precision–recall curve (PRC) [4] and average precision (AP) as the metrics to measure the performance of the detectors. PRC is plotted using the recalls and precisions on all testing data under different threshold values, and AP calculates the area under the PRC. As in [4], a detection is considered to be correct if its bounding box overlaps more than 50% with the ground truth. We also compared the proposed WSL-based framework with the supervised learning scheme by the same set of training images and features. The supervised learning scheme is implemented by the training target detector (linear SVM) based on the human annotation (labeled rectangle for each target in the training image set) and detecting targets via the scheme in Section II-D. Fig. 6 shows the PRCs and APs of different detectors. As can be seen, although the detection results differ by using various features, the proposed WSL approach can always achieve a performance comparable with the supervised method. More surprisingly, the proposed WSL approach can obtain better results in some cases. Specifically, it can improve the performance of the corresponding supervised method by 12.41% for airplane detection with the BOW feature, 11.31% for airplane detection with the LLC feature, 4.94% for vehicle detection with the BOW feature, and 6.36% for airport detection with the pHOG feature. Besides the reason that some positive training instances are missed by the human labelers, another important reason for the better performance of the proposed WSL method may be due to the iterative refinement of the detector in the training scheme. Essentially, the iterative target detector refinement is very similar to the bootstrapping methods used for training the classifier in supervised learning, which updates the training samples by certain criterions and trains the classifier in a few iterations. In addition, the multithreshold segmentation and the negative mining processing are also important factors for the performance improvement because they can generate potential positive training examples which are most distinct from the objects in negative images and likely to be the target of interest.

To demonstrate the efficiency of the proposed target detection scheme, we compared the proposed candidate-patch-based scheme (CP) with the traditional sliding-window-based scheme (SW) using the feature of pHOG. Here, we determine the sizes of sliding windows according to the potential size of targets and set the step to be 10% of the sliding window side length as suggested in [4]. The experimental results are shown in Table II, in which the average running time for each test image and the detection performance are listed. The experiments were conducted on a PC with a 3.40-GHz CPU and a 4-GB memory. As can be seen, the proposed detection scheme cannot only reduce the running time but also improve the detection performance. This is because detecting targets based on the candidate patches can decrease the search scope and the potential false positive results to a large extent.

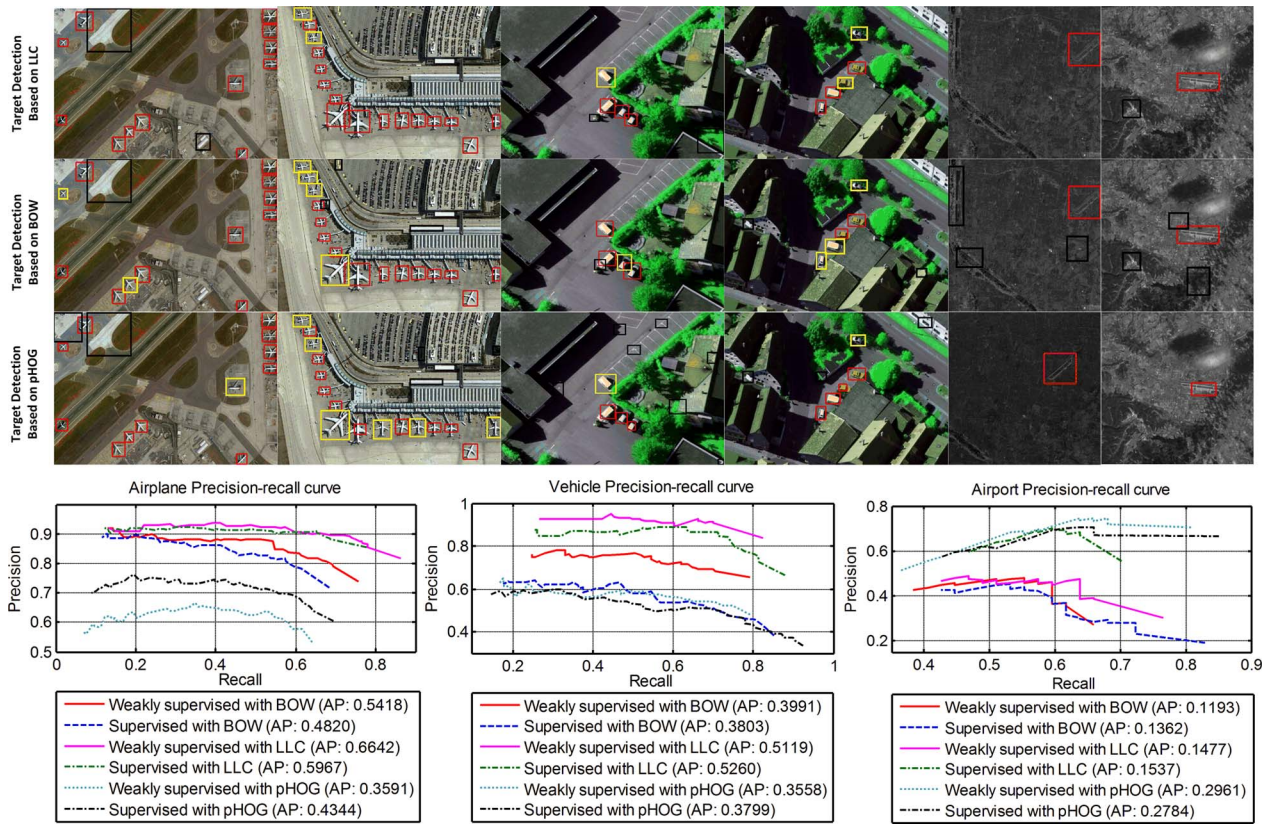

Fig. 6. Examples of detection results and the PRCs.

TABLE II
EXPERIMENTAL RESULTS FOR DIFFERENT DETECTION SCHEMES

	Google Earth		ISPRS		Landsat	
	SW	CP	SW	CP	SW	CP
Time(s)	412.67	34.4	494.80	23.23	42.53	28.74
AP	0.0551	0.3591	0.0704	0.3558	0.1257	0.2961

IV. CONCLUSION

In this letter, we have developed a framework of detecting targets in RSIs. The contributions of our work are summarized as follows: 1) A novel detection framework based on WSL techniques is proposed. It largely reduces the human labor of annotating training data and achieves a remarkable detection performance. 2) Comprehensive evaluations on three different benchmark databases were constructed, and comparisons with traditional supervised learning schemes were performed.

ACKNOWLEDGMENT

The authors would like to thank the German Society for Photogrammetry, Remote Sensing and Geo-information for providing the Vaihingen data set.

REFERENCES

[1] M. Tello, C. López-Martínez, and J. Mallorqui, "A novel algorithm for ship detection in SAR imagery based on the wavelet transforms," *IEEE Geosci. Remote Sens. Lett.*, vol. 2, no. 2, pp. 201–205, Apr. 2005.
[2] B. Sirmacek and C. Unsalan, "Urban-area and building detection using SIFT keypoints and graph theory," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 4, pp. 1156–1167, Apr. 2009.
[3] H. Sun, X. Sun, H. Wang, Y. Li, and X. Li, "Automatic target detection in high-resolution remote sensing images using spatial sparse coding bag-of-words model," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 1, pp. 109–113, Jan. 2012.
[4] J. Han *et al.*, "Efficient, simultaneous detection of multi-class geospatial targets based on visual saliency modeling and discriminative learning of sparse coding," *ISPRS J. Photogramm.*, vol. 89, pp. 37–48, Mar. 2014.
[5] Q. Liu, X. Liao, and L. Carin, "Detection of unexploded ordnance via efficient semisupervised and active learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 9, pp. 2558–2567, Sep. 2008.
[6] L. Capobianco, A. Garzelli, and G. Camps-Valls, "Target detection with semisupervised kernel orthogonal subspace projection," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 11, pp. 3822–3833, Nov. 2009.
[7] F. Zheng *et al.*, "A semi-supervised approach for dimensionality reduction with distributional similarity," *Neurocomputing*, vol. 103, pp. 210–221, Mar. 2013.
[8] P. Siva, C. Russell, and T. Xiang, "In defence of negative mining for annotating weakly labelled data," in *Proc. ECCV*, 2012, pp. 594–608.
[9] F. Zhu and L. Shao, "Weakly-supervised cross-domain dictionary learning for visual recognition," *Int. J. Comput. Vis.*, vol. 109, no. 1/2, pp. 42–59, Aug. 2014.
[10] A. Borji, "Boosting bottom-up and top-down visual features for saliency estimation," in *Proc. IEEE CVPR*, 2012, pp. 438–445.
[11] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," in *Proc. NIPS*, 2007, pp. 545–552.
[12] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *Proc. IEEE CVPR*, 2007, pp. 1–8.
[13] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *Proc. IEEE CVPR*, 2009, pp. 1597–1604.
[14] B. Han, H. Zhu, and Y. Ding, "Bottom-up saliency based on weighted sparse coding residual," in *Proc. ACM MM*, 2011, pp. 1117–1120.
[15] M. Pandey and S. Lazebnik, "Scene recognition and weakly supervised object localization with deformable part-based model," in *Proc. IEEE ICCV*, 2011, pp. 1307–1314.
[16] M. Cramer, "The DGPF test on digital aerial camera evaluation—Overview and test design," *Photogramm.-Fernerkundung-Geoinf.*, no. 2, pp. 73–82, May 2010.
[17] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. ECCV Workshop Stat. Learn. Comput. Vis.*, 2004, pp. 1–22.
[18] J. Wang *et al.*, "Locality-constrained linear coding for image classification," in *Proc. IEEE CVPR*, 2010, pp. 3360–3367.
[19] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *Proc. ACM CIVR*, 2007, pp. 401–408.