ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/327616822

Vehicle Detection in High-Resolution Images Using Superpixel Segmentation and CNN Iteration Strategy

Article *in* IEEE Geoscience and Remote Sensing Letters · September 2018

DOI: 10.1109/LGRS.2018.2866816

CITATIONS
3

READS
486

4 authors, including:

Yushi Chen
Harbin Institute of Technology
72 PUBLICATIONS **3,485** CITATIONS
SEE PROFILE

Shengwei Zhong
19 PUBLICATIONS **41** CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Feature extraction and classification of hyperspectral images based on convolutional neural network View project

Remote sensing image processing View project

All content following this page was uploaded by Shengwei Zhong on 16 October 2018.

The user has requested enhancement of the downloaded file.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Vehicle Detection in High-Resolution Images Using Superpixel Segmentation and CNN Iteration Strategy

Di Wu, Ye Zhang, Yushi Chen, and Shengwei Zhong

***Abstract*— This letter presents a study of vehicle detection in high-resolution images using superpixel segmentation and iterative convolutional neural network strategy. First, a novel superpixel segmentation integrated with multiple local information constraints method is proposed to improve the segmentation results with a low breakage rate. To make training and detection more efficient, we extract meaningful and nonredundant patches based on the centers of the segmented superpixels. For reducing the instability in detection performance because of manual or random selection of samples, a training sample iterative selection strategy based on convolutional neural network is proposed. After a compact training sample subset is obtained from the original entire training set, a representative feature set with high discrimination ability between vehicle and background is extracted from these selected samples for detection. To further avoid overfitting the training and promote the detection efficiency, data augment and a main direction estimation method are used. Comparative experimental results on Toronto data indicated the effectiveness of our proposed method.**

***Index Terms*— Convolutional neural network (CNN), high-resolution images, superpixel segmentation, vehicle detection.**

I. Introduction

BENEFIT from high-resolution images, the detection or recognition of the small man-made target becomes possible. With the increasing number of vehicles, vehicle detection methods have been studied for intelligent transportation, road network planning, and estimating parking situations to relieve the traffic jam and congestion prevention.

Many approaches have been developed for vehicle detection in high-resolution images. Generally speaking, the existing approaches mainly consist of three stages: vehicle location, feature description, and vehicle detection. For vehicle location, the sliding window method is most widely used; however, a fixed widow size or stride length seriously influences the processing speed and the precision rate of detection, which is not satisfied with the currently detection demand. As a result, many excellent segmentation approaches have been introduced, such as simple linear iterative clustering (SLIC) [1], edge-weighted Centroidal Voronoi Tessellations-based algorithm (VCells) [2], and entropy-rate clustering [3]. They have been applied on object detection work in some research. Chen *et al.* [4] proposed a superpixel segmentation and sparse representation to generate vehicles patches and extract histogram of oriented gradient (HOG) feature for detection. Yu *et al.* [5] presented rotation-invariant object detection using superpixel segmentation and Hough forests. Vehicle detection based on superpixel segmentation and fast sparse representation classification was used in [6].

Similarly, various traditional feature descriptions of vehicle are used in vehicle detection. Such as HOG + support vector machine (SVM) [7], HOG + local binary pattern (LBP) + SVM [8], and scale-invariant feature transform (SIFT) + SVM [9]. Besides these hand-crafted features, many object detection research based on convolutional neural network (CNN) features are proposed because of the excellent performance of CNN. In computer vision field, Liu *et al.* [10] used single-shot multibox detector and acquired much faster and more accurate results than [11]. Ren *et al.* [12] proposed a state-of-the-art framework (faster R-CNN), which merged region proposal network (RPN) and Fast R-CNN into a single network to detect object. In order to extend R-CNN to vehicle detection on aerial images, Deng *et al.* [13] proposed a coupled R-CNN method through combining accurate vehicle proposed network and a vehicle attribute learning network, which achieved an impressive performance. Many optimized approaches based on CNN also have been used to detect vehicles. In [14], multiscale spatial pyramid pooling-based CNN is applied to vehicle detection. Denoizing-based CNN is employed to improve the problem of training overfitting for obtaining better detection results [15]. In [16], deep learning approach is used to detect cars on the small segmented regions. Chen *et al.* [17] designed a hybrid deep CNN to detect vehicles. Two approaches [18], [19] were based on CNN and hard negative example mining for vehicle detection.

However, existing methods also suffer from the following two problems on vehicle location and feature description. For vehicle location, most methods are pixel-based or sliding window-based with fixed steps; namely, a large or small window size/stride length may miss or over-segment objects, and further increasing the compute burden or influencing the detection's precision rate. In addition, it is difficult to separate vehicles when they are parked in a region and the distances among them are small. Although superpixel segmentation methods get better performance, their segmentation results still suffer from a high breakage rate. Most detection methods which based on RPN and CNN also need to manually annotate the training samples rather than automatic selection, and they are more suitable to process visual images than aerial images with small object. For vehicle feature, many feature descriptions (such as HOG, LBP, and SIFT) belong to shallow-layer

Manuscript received February 8, 2018; revised May 11, 2018 and June 15, 2018; accepted August 17, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61471148 and in part by the Defense Industrial Technology Development Program under Grant JCKY2016603C004. *(Corresponding author: Di Wu.)*

The authors are with the Department of Information Engineering, Harbin Institute of Technology, Harbin 150001, China (e-mail: woodi87@163.com).

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

Digital Object Identifier 10.1109/LGRS.2018.2866816

1545-598X © 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

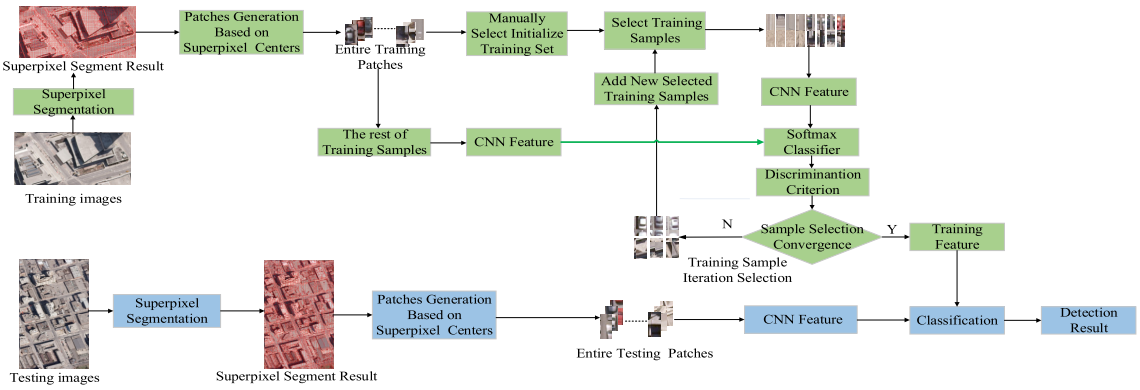

Fig. 1. Framework of our proposed method.

feature fail to get optimal feature representation for vehicle. Meanwhile, a large size of image with complex background causes a large number of negative samples, which is hard to obtain an optimal training set via manual or random selection. Consequently, an effective segmentation method and sample selection strategy based on better feature description need to be developed.

In this letter, we proposed a novel vehicle detection in high-resolution images based on superpixel segmentation and iterative CNN strategy. There are two main contributions of this letter.

1) To improve the detection efficiency and precision, we proposed a novel superpixel segmentation method combined with multiple local information constraints, which can effectively extract patches based on the centers of the segmented superpixels with low breakage rate.
2) An automatic training sample iterative selection strategy based on CNN is proposed. Both representative positive and negative samples are selected by predefine criterion without manual work. Meanwhile, the abundant deep features are extracted from samples by the predesigned CNN structure for obtaining better performance.

II. Methodological Overview of the Vehicle Detection Procedure

As shown in Fig. 1, the proposed method consists of training and detection stages. During the training stage, which mainly contains two phases, i.e., patches generation via superpixel segmentation and training sample iterative selection. In the first phase, a superpixel segmentation method combined with multiple local information constraint is used to segment all training images. Then, the meaningful and nonredundant patches are generated based on the centers of the segmented superpixels. In the second phase, a training sample iterative selection strategy based on CNN is applied. In this process, we first choose dozens of positive samples without interference (shadows, occlusions, etc.) and several kinds of negative samples as an initial sample set, and these samples are trained to acquire initial CNN model. After that, in each iteration, the classification scores of all other training samples are computed by the trained model, and the network model is then updated when new samples are added according to the predefined criterion. Once the process has converged, we use these trained features for detection.

During detection stage, the patch candidates are generated from the testing images using the above-mentioned superpixel segmentation method, and the same structure of CNN is applied to extract deep features from these patches for subsequent classification.

III. Implementation Detail

In this section, we proposed a novel vehicle detection framework using superpixel segmentation and training sample iterative selection strategy based on CNN.

A. Superpixel Segmentation With Multiple Local Information

Given an image $I = \{r(e), g(e), b(e)\}_{e\in I}$, where $r(e)$, $g(e)$, and $b(e)$ represent the three channels of image, that is, red(R), green (G), and blue (B), respectively. Then,

$$R_{Q_i} = \frac{1}{|q_i|}\sum_{e\in q_i} R(e) \tag{1}$$

$$G_{Q_i} = \frac{1}{|q_i|}\sum_{e\in q_i} G(e) \tag{2}$$

$$B_{Q_i} = \frac{1}{|q_i|}\sum_{e\in q_i} B(e) \tag{3}$$

where $Q = \{q_i\}_{i=1}^{j}$ is the partition of image I, $|q_i|$ is the number of pixels in partition q_i, j represents the initial partition number, and R_{Q_i}, G_{Q_i}, and B_{Q_i} represent each color center of the RGB channels, respectively. After initial partitions are obtained, we iteratively update each partition's boundary pixels via three constraints, that is, space distance constraint, color distance constraint, and multiple local information constraints.

The first constraint is the space distance between a boundary pixel e and a neighbor partition q_i's space center q_{is}, which is calculated as

$$d_s(e, q_i) = \sqrt{(x - x_{q_{is}})^2 + (y - y_{q_{is}})^2} \tag{4}$$

where x and y are the coordinates of a boundary pixel e in the image.

The second constraint is the color distance from pixel e to its neighbor partition q_i's color center q_{ic}, which is calculated as

$$d_c(e, q_i) = \sqrt{(r(e) - R_{q_{ic}})^2 + (g(e) - G_{q_{ic}})^2 + (b(e) - B_{q_{ic}})^2}. \tag{5}$$

The third constraint is a boundary pixel e's multiple local information, which is the statistical probability of pixels belonged to partition q_i within multiple local areas centered on pixel e, and it is computed as

$$d_p(e, q_i) = \alpha\frac{N_m(e, q_i)}{TN_m} + (1 - \alpha)\frac{N_n(e, q_i)}{TN_n} \tag{6}$$

where $N_m(e, q_i)$ and $N_n(e, q_i)$ represent the number of pixels in partition q_i within e's two local areas, and TN_m and TN_n are the total pixel number within the defined e's two local areas. m and n represent the size of local area. α denotes the

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

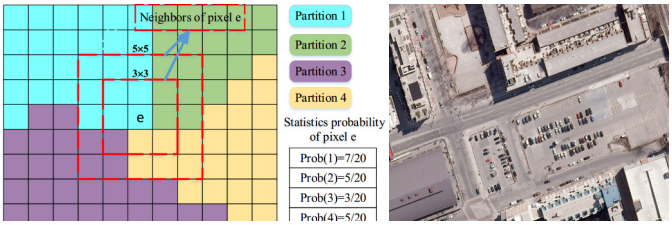

Fig. 2. (a) Statistical probability of pixel e within multiple local areas. (b) Testing image (1644 × 1152 pixels) from the Toronto data.

weighting factor. An example is shown in Fig. 2(a), the two red rectangles within 3×3 and 5×5 are defined as pixel e's local areas, and then, we can compute the statistical probabilities of pixel e in the adjacent partition. (i.e., partitions 1, 2, 3, and 4).

Finally, the probability of pixel $e \in q_i$ is calculated by

$$P(e, q_i) = \frac{S_N}{d_s(e, q_i)} \cdot \frac{C_N}{d_c(e, q_i)} \cdot d_p(e, q_i) \tag{7}$$

where S_N and C_N represent the normalization term of space and color distance, respectively. They are applied to make the smallest space and color distance between the boundary pixels and their corresponding neighbor partitions, respectively.

B. Patch Main Direction Estimation and Data Augmentation

Generally, vehicle detection via rotating each patch with an angle interval for scanning not only increases the scanning time, but also influences the detection efficiency. To solve this problem, we define the main direction of a patch as vertical; then, each patch is estimated and automatically rotated to their main direction; the more details are shown in [4]. Considering the limited size of training set, we necessarily increase the number of training samples to avoid overfitting during the training stage. First, we define n rotation angles $\varphi = \{\varphi_1, \varphi_2, \varphi_3, ..., \varphi_n\}$ and their rotation transformations $\mathrm{RT}_\varphi = \{\mathrm{RT}_{\varphi_1}, \mathrm{RT}_{\varphi_2}, \mathrm{RT}_{\varphi_3}, ..., \mathrm{RT}_{\varphi_n}\}$, where RT_φ represents a sample rotation with the angle of φ. Therefore, a new training set $\mathrm{RT}_\varphi X = \{\mathrm{RT}_{\varphi_1} x_1, \mathrm{RT}_{\varphi_2} x_2, \mathrm{RT}_{\varphi_3} x_3, ..., \mathrm{RT}_{\varphi_n} x_n\}$ can be obtained by employing rotation transformations RT_φ on all training samples $X = \{x_1, x_2, x_{3,}, ..., x_{1,}\}$. Finally, the total training samples $Y = \{X, \mathrm{RT}_\varphi X\}$ are used to train the CNN model.

C. Feature Extraction Using CNN

In order to extract effective feature, CNN is chosen as feature extractor for each patch, which is generated based on superpixel segmentation centers. In this letter, we give a specific network structure of CNN, which is designed for our application. Considering the patch size of our data, we build network that contains four blocks and three full connection layers. The number of filters is configured as 64, 128, 256, and 512 in four blocks (from first to fourth block). Each block composes of two convolution layers, one rectified linear unit layer and one max pooling layer. The kernel size of convolution and pooling layer are set as 3×3 and 2×2, respectively, and the stride and padding are set as 1, 1 in convolution layer and 2, 0 in pooling layer, respectively. In addition, dropout is employed (which is set as 0.5 in our method) in fully connected layers to prevent overfitting. In the training stage, we set parameters as follows: the batch size is 200, the learning rate ranges from 0.05 to 0.001, and the weight decay and the momentum term are 0.005 and 0.9.

D. Training Sample Iterative Selection Strategy

In this section, an automatic training sample iterative selection approach based on CNN is proposed, whose purpose is to construct optimal feature representation to improve detection performance. Initially, we manually select 50 car samples with clear textures without interference, and 500 background samples (include grass, road, and building) as an initial training set. Then, the CNN features of the selected training samples are extracted for training the initial CNN model. Second, all the other training samples are classified by the initial CNN model, and each sample is assigned a number to forbid the reselection. Third, based on the predefined criteria, we compute the classification score of each sample and add them to the training set. The criteria are defined as follows:

$$\begin{aligned} S_{\text{score}} &= \begin{cases} \text{classify} \rightarrow \text{positive}, & S_{\text{label}} \in \text{pos} \\ \text{classify} \rightarrow \text{positive}, & S_{\text{label}} \in \text{neg} \end{cases} \\ &\Rightarrow \begin{cases} S_{\text{score}} > T_{\text{pos}}, & \text{add positive sample} \\ S_{\text{score}} < T_{\text{neg}} & \text{add negative sample} \end{cases} \end{aligned} \tag{8}$$

where S_{score} represents the classification score of a sample, and S_{label} denotes the truth label of a sample.

According to (8), we can estimate the similarity between positives and negatives, and the samples are selected to join the training set guided by the following two criteria.

1) *A Positive Sample*: a new positive sample will be selected and added into training set, when it is classified as a positive one and the value of S_{score} is lower than the defined threshold T_{pos} (0.5 in our method).
2) *A Negative Sample*: a new negative sample will be selected and added into training set, when it is classified as a positive one and the value S_{score} is higher than the defined threshold T_{neg} (0.8 in our method).

The first criterion ensures the distinction of the within-class car samples, and the second criterion ensures the discriminability of car and background samples. Then, we can iteratively run these steps to select more representative training samples until our predefined convergence conditions are reached. In this letter, we terminated the training sample selection after four iterations, at that point, we obtained a classification accuracy value of greater than 82% under 0.7 recall rate.

IV. Experiment and Results

In Section IV-A, we first analyze and evaluate superpixel segmentation algorithm with Berkeley data set. Then, we test our method on Toronto data set compared with other methods.

A. Data Set Description and Experimental Configuration

1) Data Set Description: Two data sets were used in the experiments. The first data set was public Berkeley data set, which contains 500 images. It is widely used to evaluate the performance of superpixel segmentation method. The second data set was Toronto data, which covers the city of Toronto. It was a size of 11500 × 7500 pixels with a spatial resolution of 0.15 m. In our experiment, the image is cut into 13 subimages for training and eight subimages for testing, respectively. The testing images contain 1589 cars. Fig. 2(b) shows an example of testing image with complex background from Toronto data.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

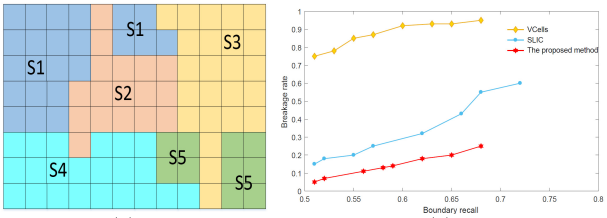

Fig. 3. (a) Example of segmentation breakage. (b) Comparison of different superpixel segmentation methods.

2) Evaluation Metrics: Boundary recall and breakage rate are used to evaluate the performance of our superpixel segmentation method. The breakage rate is defined as disconnection of segmentations. Fig. 3(a) shows a diagram of segmentation breakage. In Fig. 3(a), we assume that an image S is segmented to five parts, i.e., $S1$–$S5$. $S1$ (blue part) and $S5$ (green part) are divided into two partitions and they do not connect to each other. Therefore, the breakage rate is defined as

$$S_{\text{br}} = \frac{\ell(S)}{|S|} \tag{9}$$

where $\ell(S)$ and $|S|$ represent the number of disconnected segmentations of S and total segmentations of S, respectively.

The precision–recall curve (PRC) and F1-score are used as the evaluation metrics of Toronto data set. In detection stage, if one ground truth is redetected multiple times, only the one that exactly located on a ground truth is considered as true positive detection, and other detection results are the false alarms.

3) Compared Approaches and Parameter Configuration: To evaluate the segmentation performance, two traditional state-of-the-art segmentation methods are used for comparison, i.e., SLIC [1] and VCells [2]. The codes of VCells and SLIC were obtained from the author and VLFeat, respectively. We only adjust the parameters about superpixel number and boundary recall; other parameters are default values in the two codes.

To verify the performance of our method, six other competitive methods are tested on images for comparison, whose codes are publicly available, including HOG + SVM, SIFT + SVM, LBP + SVM, ACF detector, ACF + Fast R-CNN [13], Faster R-CNN [12], and the proposed method + random samples. In the above three SVM-based methods, a sliding window scanning procedure based on superpixel centers is used. During the training stage, 740 random (or selected) positives and 8530 random (or selected) negatives with a size of 40 × 20 pixels are used. For HOG and SIFT methods, the cell size was set as 5 × 5 pixels. For the LBP method, the cells were set as 7 × 7 pixels. The ACF detector was trained with a sliding widow size of 48 × 48 pixels and 2048 weak classifiers. For ACF+ Fast R-CNN method, ACF detector is used to first generate region of interest regions and then they are input to classify by Fast R-CNN. The above R-CNN-based methods are trained on the basic data set.

B. Superpixel Segmentation Experiment With Berkeley Data

For comparing with these methods, we predefined superpixel size as approximately 250 pixels per superpixel under boundary recall rate ranging from 50% to 70% for experiment.

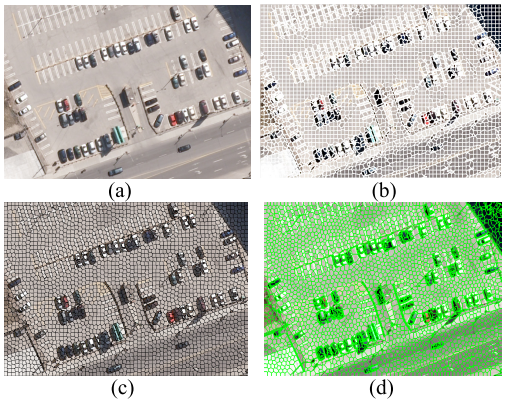

Fig. 4. Comparison of different segmentation results. (a) Original image. (b) Proposed method. (c) SLIC. (d) VCells.

Fig. 3(b) shows that the experimental results of our proposed method obtained the lowest breakage under the same boundary recall rate among the three methods. Both SLIC and VCells get high breakage rate when the boundary recall rate is higher than 0.5. In addition, we also analyze the effect of the three constraints on segmentation results in eight testing images. The average probability values of first, second, and third constraints are 0.2277–0.2313, 0.5184–0.6362, and 0.7318–0.7349; we can observe that the multiple local information constraint has the biggest effect among three constraints, and the improvements of segmentation results have proved its effectiveness.

Because of the high breakage rates of SLIC and VCells, they will generate a large number of small segmentation fragments when processing objects with complex texture, and further resulting in the increment of detection burden and false alarm rate. By comparison, we use the multiple local information to improve the segmentation process and successfully decrease the segmentation breakage rate. Fig. 4 shows a visual comparison of segmentation results including SLIC, VCells, and our method. It shows that the segmentation result of the proposed method is more regular and smoother than the other two methods.

C. Experimental Results With Toronto Data Set

In the training stage, we segmented all the training subimages into superpixels with an approximate size of 450 pixels and generated about 182 000 superpixel patches, which include 4820 car patches. Considering that the size of a car generally does not exceed 5.5 × 2.5 m^2, which means that it generally contains about 37 × 17 pixels. So we first generated patches based on superpixel centers with a size of 61 × 61 pixels. Then, these patches were rotated to their main directions and clipped to 40 × 20 pixels in order to reduce the interference of other car's or background edge information. Lastly, we resized them to the size of 48 × 48 pixels as input of CNN, because it obtained better experimental performance than the size of 40 × 20 pixels according to the experiments had been done.

To evaluate training sample iterative selection strategy, we used the trained CNN to detect vehicles on testing images. In Fig. 5(a), we can observe that the precision of detection gradually rises with the more positive and negative samples that are selected to join the training set in each iteration. It means that the trained features have the ability to separate

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

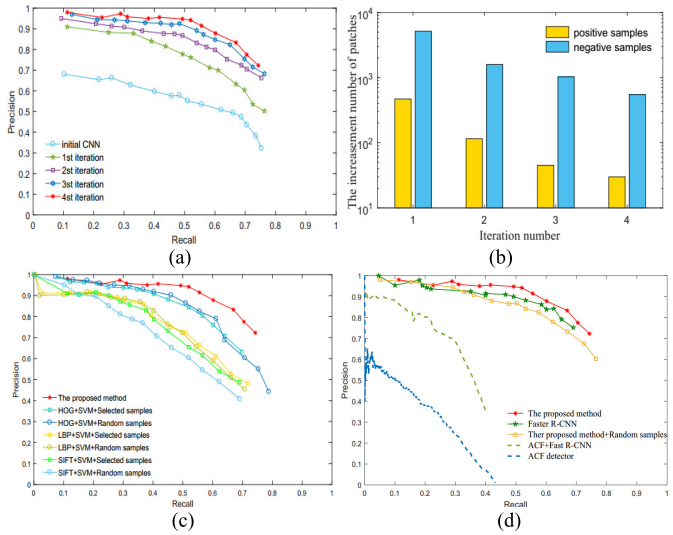

Fig. 5. (a) Performance of our method on testing images. (b) Increasing number of selected patches in each iteration. (c) Comparison with three SVM-based methods. (d) Comparison with three CNN-based methods.

the vehicles and backgrounds, when the positives that have low similarity to the selected positives and the negatives that have high similarity to the positives are added. In Fig. 5(b), we can observe that the number of selected samples gradually decreased with each iteration, and a small number of positive samples (32 cars) are selected in the last iteration, which means that nearly all representative samples have been selected to the training set after four iterations.

As shown in Fig. 5(c), HOG + SVM acquires a good performance among the three methods using randomly selected samples. The experimental results of three methods using our iteratively selected samples have similar PRCs compared with these methods using randomly selected samples. But they have dramatic decrease when the recall rate is higher than 0.4. By contrast, our method gets precision approximately 15%–25% higher than these SVM-based methods, when the recall rate is higher than 0.5. In Fig. 5(d), ACF + Fast R-CNN acquires a better result than ACF detector, because it used the Fast R-CNN as the classifier, but its precision is also lower than the proposed method + random samples under the same recall rate. Faster R-CNN achieves the best performance among these compared methods. In our case, the average size of the testing images is 2456 × 1418 pixels, and the average computation time of Faster R-CNN (10.6 s/image) is less than ours (27.5 s/image), because it only needs to propose and detect a certain number of object-like regions, rather than our method needs to scan all the image based on superpixel centers. But our method can extract all the regions (patches) that contain the true vehicle; in contrast, Faster R-CNN may miss the regions that contain the true vehicle, when the dark vehicles are small in shadow or darker complex background. Moreover, the F1-score of our method (0.74 under recall rate 70.42% and precision rate 77.54%) is higher than Faster R-CNN (0.72 under recall rate 68.97% and precision rate 75.22%); it also demonstrates that the proposed method is effective and accurate compared with other methods.

V. Conclusion

In this letter, we have presented a vehicle detection method using superpixel segmentation and training sample iterative selection strategy. By decreasing the breakage rate to improve detection efficiency, a superpixel segmentation method combined with multiple local information constraints is proposed. To construct a representative feature set, a training sample iterative selection strategy based on CNN is developed. The quantitative comparison results on two data sets have shown obviously outstanding performance of the proposed method.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[2] J. Wang and X. Wang, "VCells: Simple and efficient superpixels using edge-weighted Centroidal Voronoi tessellations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1241–1247, Jun. 2012.

[3] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, "Entropy-rate clustering: Cluster analysis via maximizing a submodular function subject to a matroid constraint," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 99–112, Jan. 2013.

[4] Z. Chen *et al.*, "Vehicle detection in high-resolution aerial images via sparse representation and superpixels," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 1, pp. 103–116, Jan. 2016.

[5] Y. Yu, H. Guan, and Z. Ji, "Rotation-invariant object detection in high-resolution satellite imagery using superpixel-based deep Hough forests," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 11, pp. 2183–2187, Nov. 2015.

[6] Z. Chen *et al.*, "Vehicle detection in high-resolution aerial images based on fast sparse representation classification and multiorder feature," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2296–2309, Aug. 2016.

[7] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jun. 2005, pp. 886–893.

[8] H. Grabner, T. T. Nguyen, B. Gruber, and H. Bischof, "On-line boosting-based car detection from aerial images," *J. Photogramm. Remote Sens.*, vol. 63, no. 3, pp. 382–396, May 2008.

[9] T. Moranduzzo and F. Melgani, "A SIFT-SVM method for detecting cars in UAV images," in *Proc. IEEE IGARSS*, Jul. 2012, pp. 6868–6871.

[10] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016.

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

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

[13] Z. Deng, H. Sun, S. Zhou, J. Zhao, and H. Zou, "Toward fast and accurate vehicle detection in aerial images using coupled region-based convolutional neural networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 8, pp. 3652–3664, Aug. 2017.

[14] T. Qu, Q. Zhang, and S. Sun, "Vehicle detection from high-resolution aerial images using spatial pyramid pooling-based deep convolutional neural networks," *Multimedia Tools Appl.*, vol. 76, no. 20, pp. 21651–21663, 2017.

[15] H. Li, K. Fu, M. Yan, X. Sun, H. Sun, and W. Diao, "Vehicle detection in remote sensing images using denoizing-based convolutional neural networks," *Remote Sens. Lett.*, vol. 8, no. 3, pp. 262–270, 2017.

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

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

[18] T. Tang, S. Zhou, Z. Deng, H. Zou, and L. Lei, "Vehicle detection in aerial images based on region convolutional neural networks and hard negative example mining," *Sensors*, vol. 17, no. 2, p. 336, 2017.

[19] Y. Koga, H. Miyazaki, and R. Shibasaki, "A CNN-based method of vehicle detection from aerial images using hard example mining," *Remote Sens.*, vol. 10, no. 1, p. 124, 2018.

View publication stats