A Plane Target Detection Algorithm in Remote Sensing Images based on Deep Learning Network Technology

Li Shuxin¹,*, Zhang Zhilong¹ and Li Biao¹

1ATR National Key Laboratory, National University of defence technology, Changsha, province of Hunan, China

*570938695@qq.com

Abstract. Plane is an important target category in remote sensing targets and it is of great value to detect the plane targets automatically. As remote imaging technology developing continuously, the resolution of the remote sensing image has been very high and we can get more detailed information for detecting the remote sensing targets automatically. Deep learning network technology is the most advanced technology in image target detection and recognition, which provided great performance improvement in the field of target detection and recognition in the everyday scenes. We combined the technology with the application in the remote sensing target detection and proposed an algorithm with end to end deep network, which can learn from the remote sensing images to detect the targets in the new images automatically and robustly. Our experiments shows that the algorithm can capture the feature information of the plane target and has better performance in target detection with the old methods.

1. Introduction

As remote sensing technology develops continuously, the resolution of the remote sensing images is very high which can provide wider observations of the detailed information and changes on the surface of the earth. Using these remote sensing images, we can develop applications on geological monitoring, city planning, environmental monitoring, precision agriculture, geographic information system and so on[1], it can also be used in spying in the military field. As the amount of remote sensing data is becoming larger and larger, it provide more and more stress on the work of manual interpretation, which pushed the technology of detecting targets automatically developed continuously. Plane target is an mobile target on the surface of the earth, it is of great value to detect the plane targets automatically. And the high resolution remote sensing images provide the possibilities to detect the plane targets automatically.

To interpretation the remote sensing targets, the old method is based on the feature of the target and develop heuristic detectors, such as the cross feature of the plane target[2][3]. But this matching method has poor stability, which can not adapt to the complex changes in the real scene. As machine learning technology develops continuously, more and more methods based on machine learning come out[4][5][6], which extract features from sliding window and train classifier to detect targets on sliding window. These methods learn from the labeled samples from the training set, which can focus on the key high level features through learning to detect targets more robustly. The problem of these method is that they need process on large amount of sliding windows, which leads to great amount of computation that they can not adapt to the applications in large number of remote sensing data processing.
Deep learning technology comes from the study of artificial neural network in the last century. In the 21st century, as the emergence of large data sets and the continuous improvement in software and hardware, there comes a new research fever on the neural network, namely deep learning network technology. The method brings more depth to the neural network with new network structure to realize the learning and processing of complex problems. The deep learning network method brings great improvement on many computer vision fields, such as object detection[7][8][9], recognition[10] segmentation[11]. They improved the performance on the complex big data sets continuously and make the methods more and more robust. Combining the new hardware, such as GPU, the methods can process images faster and faster to work in real time.

Combining the deep learning technology and remote sensing plane targets detection, we proposed a new algorithm to detect the plane targets. The method used the deep learning network to extract features from the images and get the key features through learning, which can detect the plane targets more robust and faster. It can bring great help to the automatically detection of the plane targets in the remote sensing images.

2. Related work
To detect the plane targets in remote sensing images automatically, there are 3 main methods. (1) heuristic method. This method used the key cross feature of the plane targets to match cross feature in the image in order to detect plane[2]. The shortage of the method is that it can not adapt to complex changes of the plane targets in the images, such as the change of target size, direction and brightness. (2) Visual saliency method[12][13]. This method use the difference of the low level features between the targets and background to capture the most salient region in the image, which always contain the plane targets. The method has some shortage that there always exists a lot of clutter in the ground scene, which lead to amount of wrong targets in the salient region. This bring some problems to detect the true targets. (3) machine learning method[4][5][6]. These methods extract features(such as HoG feature description) from the sliding windows in the training set to train a plane target classifier(such as SVM classifier), which can be used to the sliding windows in a new image to find if there exist a plane in the window. These methods can learn from large amount of training data to capture the robust feature of the target, which can lead to better performance. But they only use single type of feature, which may loss some information. And the large amount of sliding windows made the method slow.

We proposed a new plane targets detection method based on deep learning, which can dig deep features of the image to get better performance. The method use a network structure with many layers, such as convolution layers, pooling layers and full connection layers. There are many convolution temples to get different features, which flow through many convolution layers and pooling layers to get many feature combinations , so that we can classify the target in higher dimensional space to get better performance. This method can also realize the end-to-end training and testing. Combining the hardware of GPU to accelerate the convolution, the method can process images faster and avoid processing on large amount of sliding windows.

3. Plane target detection algorithm based on deep learning network
Using the deep learning network technology to detect plane target, the algorithm extract the features and detect the targets in a single end-to-end network. So we should design a network with proper network structure based on the goal of detecting the plane targets, which can capture the key features of the plane targets and detect them accurately. During training, we design proper structure of output layer and loss layer to make sure the training can converge and the network can be used to detect the targets accurately.

3.1. Network structure
The deep learning network technology can extract the feature of the target and detect the target in a single end-to-end network, which make the algorithm faster and more effective. The network structure diagram is shown in figure 1.
The first part of the network is the input part which put the image data and annotations to the network. The second part is the feature extraction part, which contains convolution layers and pooling layers. This part can extract many kinds of features using many kinds of convolution temples and concentrate the features through pooling layers. After the processing, the dimension of the feature space can be very high so that it can express the features of the target more accurate and comprehensive. In our work, we use the feature extraction part of the VGG16 network [14] as our feature extraction part, which shows great feature extraction capability in our experiments.

After the feature extraction part, there will usually be full connection layers to do more processing and special output layers for special tasks. For the target detection task, we use full connection layers to combine the high dimensional features and distinguish the features to help detection. Then we use the class output layer and the position output layer to output the target class (foreground or background) and the position of the target in the image. To ensure the algorithm detecting the target more accurately, we should divide the image into grids and use each grid to predict the final target position and class. The grids are divided follow the rules below: (1) the center of each grid represent the different position of the image and divide the image into grids of 8x8 and 4x4; (2) for each position, add new boxes of different sizes and aspect ratio so that a box can match the groundtruth target properly. During the training, we only focus the loss generated from the grid matched to a groundtruth target, and propagate back the loss to update the weights until the network training reached convergence. Figure 2 shows the groundtruth target location and the corresponding divided grids.

![Network structure diagram.](image-url)
Figure 2. The groundtruth target location and the corresponding divided grids

In our work, to detect the targets in more sizes, there are more convolution layers and pooling layers to produce feature maps of different sizes, which will each predict target class and position. The output layer contains class output layer (softmax layer) and position output layer.

3.2. Network training

After designing the network structure, we should also choose the method of loss computation and the strategy of network weight update, so that the network will reach convergence faster and more effectively.

To compute the loss, we should compute the loss of the class output layer and the position output layer. Using $x^p_{ij} = \{1, 0\}$ to represent that the $i$th divided grid is matched to the $j$th groundtruth target in class $p$. So all the loss is the weighted sum of the class loss and the position loss:

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)) \quad \text{MERGEFORMAT} (1)$$

In which, $N$ is the number of the matched divided grids. If $N=0$, we set the loss to 0. The parameter $\alpha$ usually set to 1.

We use the Smooth L1 loss function to calculate the loss of position, which calculated between the parameters of the predict box ($l$) and the groundtruth box ($g$).

$$L_{loc}(x, l, g) = \sum_{i=Pos} \sum_{m\in\{cx, cy, w, h\}} x^i_{jm}\text{smooth}_{l}(l_i - g_i)^2 \quad \text{MERGEFORMAT} (2)$$

$$\hat{g}^{cx} = (g^{cx}_i - d^{cx}_i) / d^{cx}_i, \hat{g}^{cy} = (g^{cy}_i - d^{cy}_i) / d^{cy}_i \quad \text{MERGEFORMAT} (3)$$

$$\hat{g}^{w} = \log\left(\frac{g^{w}_i}{d^{w}_i}\right), \hat{g}^{h} = \log\left(\frac{g^{h}_i}{d^{h}_i}\right) \quad \text{MERGEFORMAT} (4)$$

As the output of the predict $l$ is the offset of $(cx, cy, w, h)$, so we should calculate the offset $\hat{g}$ between the groundtruth information $g_i$ and the divided grid information $d_i$.

The loss of the class output is calculated using the softmax loss function of $c$ classes:

$$L_{conf}(x, c) = - \sum_{i=Pos} x^p_{ij} \log(\hat{c}^p_i) - \sum_{i=Neg} \log(\hat{c}^p_i), \hat{c}^p_i = \frac{\exp(c^p_i)}{\sum_p \exp(c^p_i)} \quad \text{MERGEFORMAT} (5)$$

To update the network weights, we use the gradient descent method. We set the initial learning rate to 0.001 and reduce it by 10 times every 5000 iterations.

In our experiments, we showed that the training can reach the convergence effectively and obtain good performance using the parameters above.

3.3. Network test

In our testing, we used the remote sensing images of big size. In order to control the performance loss of different size, we cut the images into small patches to process in the network. Finally, we combined
the detection results and processed using the non-maximum suppression to get the final target detection results.

4. Experiments and analysis

4.1. Image sets description
The image sets used in the experiments contain colour remote sensing images from google earth and some panchromatic remote sensing images. The colour remote sensing images contain plane target from different airport background and different illumination. The panchromatic remote sensing image are gray image with big size. The resolution of the images are around 1m. We divide the images into 3 sets: colour image set 1 with 1190 images; colour image set 2 with 200 images; gray image set with 4 images. The sample images are showed in figure 3.

![Image](image1.png)  ![Image](image2.png)

Figure 3. Remote sensing images in the datasets

4.2. Experiment result and analysis
We used the 200 images randomly selected from the colour image set 1 to train the deep network and used another 200 randomly selected images as the verification data. We trained 20000 iterations to get the final model and used it to test on the 3 image sets, colour image set 1 with 1190 images; colour image set 2 with 200 images; gray image set with 4 images. The test results are shown in figure 4.

![Image](image3.png)  ![Image](image4.png)  ![Image](image5.png)

(a)  (b)  (c)
Figure 4. Plane target detection result

In figure 4, figure (a)-(f) shows the result of images from colour image set 1, figure (g)-(h) shows the result of images from colour image set 2, figure (i)-(j) shows the screenshot result of images from gray image set. The results show that the algorithm can detect the plane targets in complex background effectively without the influence of changes in target size, shape, orientation and luminance, and the false alarm rate is low.

After the subjective evaluation, we analysed the objective evaluation of the results quantitatively. The main evaluation indicators for target detection is Recall, Precision, false alarm rate and AP(Average Precision). Recall is the ratio of the number of the targets detected correctly and the number of the groundtruth targets; Precision is the ratio of the number of the targets detected correctly and the number of the targets detected; false alarm rate is the ratio of the number of the targets detected incorrectly and the number of the targets detected, which can be calculated by 1-Precision. If the Recall and Precision is higher and the false alarm rate is lower, the performance of the algorithm is better. To calculate the above 3 indicators, we need to set the threshold of the output score. AP calculates the area below the Precision-Recall curve, which is a better indicator that can reflect the comprehensive performance under different threshold of the output score. When the AP is higher, the performance is better.

In the table 1, we shows the evaluation indicators of the test results on 3 different image sets. The threshold used to calculate the Recall, Precision and False alarm rate is set to 0.25.
Table 1. Objective evaluation of the test results.

| Performance indicator         | Recall | Precision | False alarm rate | AP  |
|------------------------------|--------|-----------|------------------|-----|
| Colour image set 1           | 0.907  | 0.876     | 0.124            | 0.806|
| Colour image set 2           | 0.852  | 0.601     | 0.399            | 0.760|
| Gray image set               | 0.577  | 0.608     | 0.392            | 0.634|
| Mean                         | 0.779  | 0.695     | 0.305            | 0.733|

From the objective evaluation showed in the table 1, we can see that the best performance is the results tested on the colour image set 1, of which the feature distribution is most close to the training images. The performance of the colour image set 2 is still high but a little drop, as there are some plane with special shape which has big difference from the plane in the training set. The performance of the gray image set is also a little lower, as there are big amount of background in the big size image which leads to some false alarms. The mean AP is 0.733 in all the test data, which shows the overall performance is good.

4.3. Results compared with other methods

In this part, we compare the performance of our method with the conventional method, such as the method based on saliency and the method based on machine learning. In the saliency method, we calculate the difference between the low level features of different regions to get the saliency map. Then we combine the meanshift segmentation results to get the saliency region. From the saliency region, we analyse the region feature to get the final target region. In the machine learning based method, we first extract HoG features on the sliding windows of the images in training set and train a SVM(support vector machine) classifier, then we test a new image through predicting every sliding window on the image to get the final detection result. The evaluation results of the 3 method on the colour images set 1 are shown in table 2.

Table 2. Objective evaluation of the test results of 3 different methods.

|                       | Recall | Precision | False alarm rate | AP  |
|-----------------------|--------|-----------|------------------|-----|
| Saliency method       | 0.661  | 0.352     | 0.648            |     |
| HoG+SVM               | 0.671  | 0.483     | 0.517            | 0.464|
| Our method            | 0.907  | 0.876     | 0.124            | 0.806|

The results show that our method has better performance on the 4 objective evaluation indicators, which means our method has a better detection performance. In the saliency method, it only extract the low level features of the image to detect the target without combining the high level knowledge, which will consequentially detect more false alarm targets and miss more targets. In the machine learning method using HoG features and the SVM classifier, it can learning from the large numbers of image samples in the training set to extract the high level features of the target, but the performance is limited as it only uses the single HoG feature. In our method based on deep learning network, it uses the deep network to extract more features and feature combinations, which can lead to focusing on the key character of the target through learning and reach a more robust performance.

5. Conclusion

This paper has proposed a plane targets detection method based on deep learning network. The method can detect the plane target faster and more robust as it effectively used the end-to-end deep network. The experiments show that the method can extract more comprehensive and rich features from the image to get the key character though leaning, which make the method detecting the plane targets faster and more robust. The shortcoming of the method is the need of a large amount of training target images to train the deep network, which can’t be used to the noncooperative targets with little image samples. So it’s a good research direction to detect targets with fewer training image samples.
References
[1]. Cheng G and Han J 2016 Isprs Journal of Photogrammetry & Remote Sensing. 117 11-28
[2]. Alatan A and Arslan D 2012 SPIE Defense, Security, and Sensing. Vol.8398 839808-839808-7.
[3]. Cai H and Su Y 2005 Proceedings of Spie the International Society for Optical Engineering. 5985 529-34.
[4]. Cheng G, Han J, Guo L, Qian X, Zhou P, Yao X and Hu X 2013 Isprs Journal Of Photogrammetry And Remote Sensing. 85 32-43.
[5]. Zhang W, Sun X, Wang H and Fu K 2015 Isprs Journal of Photogrammetry & Remote Sensing. 99 30-44.
[6]. Cheng G, Han J, Zhou P and Guo L 2014 Isprs Journal of Photogrammetry & Remote Sensing. 98 119-32.
[7]. Girshick R, Donahue J, Darrell T and Malik J 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR. 580-7.
[8]. Ren S, He K, Girshick R and Sun J 2015 Annual Conference on Neural Information Processing Systems, NIPS. 91-9.
[9]. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y and Berg A 2016 14th European Conference on Computer Vision, ECCV. 21-37.
[10]. Krizhevsky A, Sutskever I and Hinton GE 2012 26th Annual Conference on Neural Information Processing Systems, NIPS. 1097-105.
[11]. Long J, Shelhamer E and Darrell T 2015 Computer Vision and Pattern Recognition. 3431-40.
[12]. Han J, Zhou P, Zhang D, Cheng G, Guo L, Liu Z, Bu S and Wu J 2014 Isprs Journal Of Photogrammetry And Remote Sensing. 89:37-48.
[13]. Li W and Pan C 2011 Communications in Computer & Information Science. 327-33.
[14]. Simonyan K and Zisserman A 2014 Computer Science. arXiv preprint arXiv:1409.1556.