Airport Detection on Remote Sensing Images Using Faster Region-based Convolutional Neural Network

M M Zhu1,*, Y L Xu1, S P Ma1 and H Q Ma1
1Air Force Engineering University, Xi’an, China
E-mail: *ming_paper@163.com

Abstract. An airport detection method on remote sensing images based on transfer learning and hard example mining is proposed. We use Faster region-based convolutional neural network framework with end-to-end advantage instead of the sliding windows and artificial features in the existing traditional methods. Transfer learning is used to solve this problem that the number of airport remote sensing data itself is limited. Hard example mining is used to make full use of hard examples. As a result, it can improve the ability to discriminate objects and make training more efficient. The experimental results show that the proposed method can detect different types of airports accurately under complex backgrounds. Compared with other airport detection methods, the proposed method has higher detection rate 92.6%, lower false-alarm rate 15% and shorter average processing time 0.2s.

1. Introduction
The extensive acquisition and use of high-resolution remote sensing images have greatly promoted the development of airport detection in remote sensing images. Due to the complex backgrounds and different shapes of airport, the real-time and accuracy of airport detection is faced with significant challenges. Traditional methods are mainly divided into two kinds: edge line [1] and regional segmentation [2]. The former has the characteristics of high speed and low complexity, but it is easy to be disturbed. The performance of the latter is improved but there is a problem that the sliding window overlaps redundancy. Recently, the deep learning has provided new ideas for it: many scholars use the convolutional neural network (CNN) to identify airports and runway areas [3]. However, the above methods only use CNN’s powerful identification ability. The acquisition of airport proposal boxes still depends on the manual methods (edge line and regional segmentation), the limitations of traditional methods still exist.

The design of the region-based convolutional neural networks (R-CNN) [4] framework has made a huge breakthrough in the object detection and opened the upsurge of object detection based on deep learning. However, due to the large amount of hard disk space required by training and feature repeated extraction, its speed of training and testing is very slow. Therefore, the successor of R-CNN, Fast R-CNN, is proposed [5], which greatly improve the accuracy and speed. Due to Selective Search algorithm (SS) [4], the detection speed is still very low. Faster R-CNN [6] adopts regional Proposal Network (RPN) to extract the proposal boxes, while RPN and the subsequent detection network, Fast R-CNN, share the convolution layer, which greatly improves the speed of object detection. The training of models based on deep learning often requires lots of learning samples, while the number of remote sensing images about airport is very limited. It’s difficult to train network models directly.
However, the theory of transfer learning [7] provides an effective means to solve the airport detection problem with limited data.

Therefore, this paper abandons the previous method of sliding window plus features extracted manually and takes the Faster R-CNN as the basic framework. We reuse the model trained from other natural image databases after corresponding modification and improvement. At the same time, hard example mining is used to make training more efficient.

2. Methodology

2.1. Overview of Faster R-CNN
As shown in Figure 1. Faster R-CNN network takes as input an entire image and as output predicted probability value and detection boxes. The network is mainly divided into two parts: region proposal networks (RPN) and a detection network with region of interest (RoI) pooling layer, fully connected layers and loss layers. The shared convolutional layers are a common part of two networks. Since the object detection is unified into a deep network framework, the whole framework has end-to-end characteristics.

![Figure 1. The architecture of Faster R-CNN.](image)

2.1.1. Region Proposal Network. Region Proposal Network (RPN) uses a 3×3 sliding window to convolve the feature map. Each sliding window is mapped to a lower dimensional feature. A box-regression layer and a box-classification layer take the feature as input, as shown in Figure 2.

![Figure 2. The structure of RPN.](image)

In order to meet the multi-scale objects, the proposal boxes of multiple scale and aspect ratio are considered. These proposal boxes are parameterized by anchors. 3 scales and 3 aspect ratios are used at each position.

To train RPNs, a binary class label is assigned to each anchor. A positive label is assigned to two kinds of anchors: the anchor whose Intersection-over-Union (IoU) overlap with a ground-truth box is the highest or the anchor whose IoU overlap with any ground-truth box is higher than 0.7. A negative...
label is assigned to an anchor whose IoU ratio is lower than 0.3 for each ground-truth box. The multi-task loss is used in the training, the definition of loss function is described in literature [6].

A bounding-box regression layer is used to predict a new bounding box for the detection. As shown in Figure 3, the green box indicates the ground-truth bounding box, the yellow box indicates the proposal bounding box and the red box indicates the predicted box after the regression. Specific details can be found in literature [4].

Figure 3. Bounding-box regression.

2.1.2. Non-maximum Suppression. Some proposal boxes highly overlap with each other. To reduce redundancy, non-maximum suppression (NMS) is adopted based on their scores. The specific implementation process of NMS is:

1) The classification score of the same target is ranked from high to low.

2) Select the proposal box with the highest score and calculate the IoU value with other proposal boxes. If the IoU exceeds the threshold, then reject the proposal box with lower score.

3) Select the highest score from the remaining proposal boxes and repeat step (2).

4) Repeat step (3) until all proposal boxes are traversed. The above process is essentially an iterative-ergodic-elimination process. We can set the threshold of IoU to significantly reduce the number of proposal boxes.

2.1.3. RoI Pooling Layer. To reduce the calculation redundancy, the region of interest (RoI) pooling layer matches the spatial location of the proposal region and the corresponding feature map from the same input image, and it does not repeatedly input to the same network for the calculation. At the same time, due to the different sizes of the feature maps corresponding to proposal regions but the subsequent full-connection layer requires a fixed-size input, the RoI pooling layer uses max pooling to convert the feature maps with different sizes into a fixed-size feature map.

In order to share computation between proposal regions, we sample stochastic gradient descent (SGD) mini-batches hierarchically. For each mini-batch $B$, $N$ images are sampled and then $B/N$ proposal regions are sampled from each image.

2.2. Transfer learning

In order to extract image features efficiently, the manual design features with limited capability and poor robustness are discarded, and the convolutional neural network is used to extract features. Although the number of airport remote sensing data itself is limited, there are the common low-level and intermediate-level visual features between remote sensing images and natural images. So transfer learning can solve this problem through large database pre-training and small database fine-tuning. We use the pre-training network VGG16 for ImageNet classification to initialize the weights of the CNNs and then use the airport database for fine-tuning training. The experiment finally proved that it is very effective for transfer learning.

In order to match the proposed model with airport detection, according to the characteristics of the size and shape of the airport and the experimental verification, we set 3 scales and 3 aspect ratios in our network, as shown in Table 1:
Table 1. The settings of anchor.

| Anchor scales              | Aspect ratios |
|----------------------------|---------------|
| (128², 256², 384²)        | (1:1, 2:3, 3:2) |

2.3. Hard example mining
Datasets always contain most of the easy examples and a few hard examples. Simple examples have little significance for training, while complex examples with high diversity and loss have a greater impact on classification and detection. Making full use of complex examples can improve the ability to discriminate objects. In fact, the imbalance between easy examples and hard examples is not a new challenge. The idea of dataset bootstrapping [8], typically called hard example mining, has been applied to the training of most detection tasks. Even the recent object detection model A-Fast-RCNN [9] that generates examples with occlusions and deformations using generative adversarial nets can be regarded as an example mining method. For airports with a small amount of data, hard example mining becomes more important. Therefore, referring to the idea of hard example mining, the overall structure of proposed method is shown in Figure 4:

The original detection network is duplicated into two networks, which are represented simply by the network A and B respectively and share the network parameters. The network A only has forward pass, while the network B is a standard detection network that has both forward and backward passes. Firstly, the network A takes as input the feature maps corresponding to proposal regions and performs a forward and computes loss values of all input proposal regions. Then the hard example sampling module selects \( \frac{B}{N} \) hard examples for which the current network performs worst by sorting the input proposal regions by loss. Finally, the above hard examples are used as input to the network B for normal model training. The input of network A is all the proposal regions \( P \) from \( N \) image instead of mini-batch \( B \), while the input of network B is mini-batch \( B \). We use \( N=2 \) (which results in \( P \approx 4000 \)) and \( B = 128 \).

2.4. Training
The entire networks are implement using the Caffe framework. The shared convolution layers are initialized by a pre-training model, the weights of other layers are initialized randomly with a zero-mean Gaussian distribution with standard deviation 0.01, we use a basic learning rate of 0.001, a momentum of 0.9 and a weight decay of 0.0005. We use alternating optimization strategy to train that allows for sharing convolution layers between the RPN and detection network, which is as follows:

1. The RPN is initialized with a pre-training model VGG16, the weight parameters are fine-tuned end-to-end.
2. The detection network is also initialized by the pre-training model VGG16 and trained by using the proposal regions of step-1 RPN.
3. We use the step-2 detection network to initialize RPN, fix the shared convolutional layers and only fine-tune the layers of RPN.
4. The shared convolution layers are fixed, other layers of the detection network are fine-tuned again.

3. Experiment
The proposed method is trained and tested in real remote sensing images that cut from Google Earth from all over the world. The sizes of all the images are 1000×600. The data set includes 600 images, in which 400 images contain airports and the remaining images do not contain airport but mainly background scenes such as railways, highways, bridges, and buildings. 240 airport images and 120 non-airport images are randomly selected to form a training set, and the remaining images are composed of a test set. The images are horizontally flipped with probability 0.5 to enhance the data. The computing platform consisted of a CPU of Intel i7-7700 and a GPU of NVIDIA GTX1080.

3.1. Detection results
Some detection results are shown in Figure 5, which includes various airport background types such as roads, rivers, and mountains. The second row of Figure 5 shows some airport images with unique shapes. The results prove that our model is robust to complex backgrounds and various airport structures.

![Figure 5. Detection results.](image)

3.2. Comparison with other methods
To demonstrate the efficiency of our method, the existing airport detection methods are selected for comparative experiments. The detection rate (DR), false-alarm rate (FAR) and average processing time of each method on the test set are recorded and compared. All the results in the experiment were the average of multiple random experiments, as shown in Table 2. The literature [10] is an edge-based airport detection method, the literature [11] is an airport detection method based on regional segmentation, and the literature [3] is an airport detection method based on convolutional neural network.

| Method          | DR /% | FAR /% | Average processing time /s |
|-----------------|-------|--------|---------------------------|
| literature [10] | 65    | 22.5   | 2.4                       |
| literature [11] | 71.8  | 27.5   | >100                      |
| literature [3]  | 81.1  | 35     | 21.3                      |
| Ours            | 92.6  | 15     | 0.2                       |

According to the Table 2, the proposed method has the highest detection rate, the lowest false-alarm rate and the shortest average processing time. The reason is that the previous methods adopt the
way of sliding window plus manual design feature, the calculation efficiency of sliding window is very low and the robustness of manual design features is not strong. The proposed method integrates feature extraction, acquisition of proposal boxes and object classification and regression into an end-to-end deep network framework, makes full use of CNN's powerful feature expression capabilities and allows for sharing convolutional layers between the RPN and detection network. Therefore, the proposed method has the best performance.

4. Conclusion
This paper proposes an airport detection method based on transfer learning and hard example mining. In this method, RPN and the non-maximum suppression are used to extract proposal boxes. The RoI pooling layer avoids extracting features repeatedly. Transfer learning is used to solve this problem that airport data is limited. Hard example mining is used to make full use of hard examples and make training more efficient. Both RPN and detection network share the convolution layer, which greatly improves the efficiency of airport detection. The results show that the proposed method is better than other methods. It has strong theoretical and practical significance for the real-time and accuracy of airport detection.

References
[1] Budak U, Halici U and Sengur A 2016. Efficient Airport Detection Using Line Segment Detector and Fisher Vector Representation. J. IEEE Geoscience and Remote Sensing Letters. 13(8) 1079-83
[2] Zhu D, Wang B and Zhang L 2015. Airport target detection in remote sensing images: A new method based on two-way saliency. J. IEEE Geoscience and Remote Sensing Letters. 12(5) 1096-1100
[3] Zhang P, Niu X and Dou Y 2017. Airport detection on optical satellite images using deep convolutional neural networks. J. IEEE Geoscience and Remote Sensing Letters. 14(8) 1183-87
[4] Girshick R B, Donahue J and Darrell T 2014. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. J. CVPR. 580-587
[5] Girshick R B 2015. Fast R-CNN. J. CVPR. 1440-48
[6] Ren S Q, He K M and Girshick R 2017. Faster R-CNN: towards real-time object detection with region proposal networks. J. IEEE Transactions on Pattern Analysis and Machine Intelligence. 39(6) 1137-49
[7] Pan S J, and Yang Q 2010. A Survey on Transfer Learning. J. IEEE Transactions on Knowledge and Data Engineering. 22(10) 1345-59
[8] Felzenszwalb P F, Girshick R B and Mcallester D A 2010. Object Detection with Discriminatively Trained Part-Based Models. J. IEEE Transactions on Pattern Analysis and Machine Intelligence. 32(9) 1627-45
[9] Wang X, Shrivastava A and Gupta A 2017. A-Fast-RCNN: Hard Positive Generation via Adversary for Object Detection. J. CVPR.
[10] Qu Y, Li C and Zheng N 2005. Airport Detection Base on Support Vector Machine from A Single Image. J. Fifth International Conference on Information, Communications and Signal Processing. 546-549
[11] Tao C, Tan Y and Cai H 2011. Airport Detection from Large IKONOS Images Using Clustered SIFT Keypoints and Region Information. J. IEEE Geoscience and Remote Sensing Letters. 8(1) 128-132