*remote sensing*

MDPI

*Article*

# Airport Detection Using End-to-End Convolutional Neural Network with Hard Example Mining †

**Bowen Cai [1,2], Zhiguo Jiang [1,2], Haopeng Zhang [1,2,*], Danpei Zhao [1,2] and Yuan Yao [1,2]**

1 Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China; bowencai@buaa.edu.cn (B.C.); jiangzg@buaa.edu.cn (Z.J.); zhaodanpei@buaa.edu.cn (D.Z.); yaoyuan@buaa.edu.cn (Y.Y.)

2 Beijing Key Laboratory of Digital Media, Beihang University, Beijing 100191, China

* Correspondence: zhanghaopeng@buaa.edu.cn; Tel.: +86-10-8233-8061

† This paper is an extended version of our paper published in Bowen Cai, Zhiguo Jiang, Haopeng Zhang, et al. Training Deep Convolution Neural Wetwork with Hard Example Mining for Airport Detection. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2017), Fort Worth, TX, USA, 23–28 July 2017.

Received: 10 October 2017; Accepted: 18 November 2017; Published: 21 November 2017

**Abstract:** Deep convolutional neural network (CNN) achieves outstanding performance in the field of target detection. As one of the most typical targets in remote sensing images (RSIs), airport has attracted increasing attention in recent years. However, the essential challenge for using deep CNN to detect airport is the great imbalance between the number of airports and background examples in large-scale RSIs, which may lead to over-fitting. In this paper, we develop a hard example mining and weight-balanced strategy to construct a novel end-to-end convolutional neural network for airport detection. The initial motivation of the proposed method is that backgrounds contain an overwhelming number of easy examples and a few hard examples. Therefore, we design a hard example mining layer to automatically select hard examples by their losses, and implement a new weight-balanced loss function to optimize CNN. Meanwhile, the cascade design of proposal extraction and object detection in our network releases the constraint on input image size and reduces spurious false positives. Compared with geometric characteristics and low-level manually designed features, the hard example mining based network could extract high-level features, which is more robust for airport detection in complex environment. The proposed method is validated on a multi-scale dataset with complex background collected from Google Earth. The experimental results demonstrate that our proposed method is robust, and superior to the state-of-the-art airport detection models.

**Keywords:** airport detection; hard example mining; convolutional neural network; region proposal network

## 1. Introduction

With a rapid development in sensor and remote sensing technologies, the amount of available satellite imagery has an explosive growth in recent years. It is significant to automatically access the valuable information from the huge volume of the remote sensing data [1–7]. Objects in remote sensing images (RSIs) have many different orientations, size, and illumination densities since RSIs are taken from the upper airspace with different imaging conditions. Airports detection from optical remote sensing images has attracted increasing attention, since airport is one of the most important traffic facilities with its civil and military applications. However, airport detection is still a challenging problem, because the background around the airport is much more complicated. In addition, the size of airport varies greatly with the development of the air transportation. Moreover, there is a great

imbalance between the number of airports and backgrounds in large-scale RSIs, which may lead to a biased training on detectors.

Currently, airport detection in large optical remote sensing images can be divided into two main stages. The first stage is to locate airport candidate regions with heuristic search strategy or weak classifiers, based on airport-specific geometrical property. The second stage is to extract discriminative features to depict the candidate regions obtained from first stage and train a sophisticated classifier to identify whether the candidate regions are airports or backgrounds. Owing to the generation of airport candidate regions, it is possible to accomplish a significant speed-ups and reach high object recall with considerably fewer detection windows than using sliding window paradigm. It is also allowed for developing more sophisticated classifiers on airport identification stage to achieve better airport detection performance. An effective candidate region extraction algorithm may also improve final detection quality by reducing spurious candidate regions.

Since the airport runways often have specific lengths and widths, a great number of region extraction methods use the geometric characteristics of the runway to filter the candidate regions. Kou et al. [8] utilized the line segments detector (LSD) to extract the straight line segments and analyzed the line segments to locate the airport candidate regions. In addition, Budak et al. [2] found that LSD algorithm detects large number of lines with various lengths, which interrupts the extraction of airport candidate areas. Therefore, a more robust LSD method was proposed to fetch the parallel lines of runways in RSIs, thus enhancing the detection accuracy of the airport runway. Other studies on the analysis of runways parallel lines utilized the Hough transform [9], the Radon transform [10] to extract an airport candidate regions. In order to improve the efficiency of candidate region extraction in large-scale RSIs, visual attention mechanism is employed in the airport detection, which assumes that the airport is a salient region. Zhang et al. [11] proposed a spatial-frequency visual saliency analysis algorithm to efficiently extract an airport candidate region in board-area RSIs. Taking multi-resolution into consideration, Zhao et al. [1] presented a hierarchical reinforcement learning saliency model to extract airport regions. In [12], Zhu et al. located the airport candidate region based on bottom-up graph-based visual saliency (GBVS) and top-down saliency. In [13], Yao et al. combined the cues of contrast and the line density to build the saliency model for rapidly localizing the airport candidate areas.

After candidate region extraction, it is indispensable to identify these regions and remove the spurious targets. Previous works first utilized geometric features, gradient-based features and texture-based features to characterize candidate regions, then adopted more sophisticated classifiers to precisely recognize the airport from the cluttered background. Tao et al. [14] identified airports via several statistical measures such as mean, variance, and higher order moments in the candidate regions. Zhu et al. [12] first used the scale-invariant feature transform (SIFT) [15] descriptor to characterize the regions of interest (ROIs). Then an SVM and hierarchical discriminant regression tree were used to determine whether ROIs contain an airport. In the work of [2], fisher vector was employed to encode the SIFT descriptors extracted from candidate regions and classified by an SVM to confirmed the airport areas. Yao et al. [13] used the conditional random field (CRF) model to capture the context information in ROIs for the fine detection. Zhao et al. [1] adopted Latent Dirichlet Allocation (LDA) topic model at multi-level to judge the state of the saliency detection and recognize the airports.

The majority of airport detection methods utilize line features of the runway to extract candidate regions, and concentrate on the low-level manually designed features to describe the geometric structure of airports. These methods have the advantages of simple design and easy explanation. However, the line features may cause confusions with roadways, coastline, and mountain ridge. It is also unstable to describe an airport in complex and cluttered background using manually designed features. With the excellent performance of convolutional neural network (CNN) for object detection in natural scene images [16–20], CNN has been introduced into the field of target detection at RSIs [21–27]. Zhang et al. [25] identified airport candidate regions with a CNN model transferred from AlexNet [16]. Although the satisfactory results have been reported in [25], CNN model is only used to classify the

ROIs. The ROIs extraction still relies on the manually designed features, which has a great influence on the accuracy of airport detection.

Moreover, for airport detector training in large-scale RSIs, the training set is distinguished by a serious imbalance between the number of airports and background examples, which leads to inefficient training process and even reduces final detection quality. Therefore, there is a great challenge on how to select airport and background examples to keep airport-background class balance and further train more superior classifier in a limited time.

Recent research indicates that detection datasets contain an overwhelming number of easy examples and a small number of hard examples. So many scholars tried to mine the more representative hard examples to balance the proportion of foreground-background class examples. Some promising works were reported in [23,28–30]. Cheng et al. [28] adopted a two-step iterative training strategy that alternates between updating the detection model given the training set, and selecting hard negative examples using a updated detection model. In order to reduce the spurious targets in candidate regions and improve the accuracy of the region proposal network, Zhang et al. [23] used the output score map of the region proposal network to collect the hard negative examples and employed an iterative process for retraining the network. Traditional methods usually freeze the model to mine negative examples. However, the positive example mining is also important for avoiding missing detection. In addition, freezing the model to collect hard examples would dramatically slow the progress of model training.

To tackle these problems above, especially for balancing airport-background class examples, we propose a hard examples mining based CNN method (HEM-CNN) in this paper. The proposed method combines feature extraction, region proposal generation and airport identification into a unified framework, thus accomplishing a significant speed-ups. The cascade strategy between region proposal generation and airport identification greatly inhibits the false alarms existed in airport detection. In our method, the shared CNN layers are first employed to extract a high-level feature, which makes the extraction of airport candidate regions no longer rely on manual designed features. Then, region proposal network with HEM achieves high-quality region proposals based on the high-level feature. Finally, object detection network with HEM identifies the region proposals whether contain an airport using shared CNN features. For balancing airport-background class examples in network training, we develop a new designed hard example mining layer that automatically selects hard examples by their loss. Moreover, we also define a new loss function both for region proposal network and object detection network, which uses a weight-balanced strategy to compute the loss respectively on airport and background examples. By adopting the hard example mining and weight-balanced strategy, our method releases the ratio constraints on airport and background examples in the mini-batch. This strategy avoids the network trapping into the local optimal, and improves the training efficiency.

The major contributions of our approach are summarized as follows.

1. We design a novel HEM-CNN framework for airport detection. It integrates shared features extraction, proposal detection and object localization to achieve a speed-accuracy trade-off. In addition, a more robust high-level feature is utilized to extract high-quality proposals in our framework.
2. We introduce a new designed hard example mining layer into CNN, which is efficient to improve the network training by automatically select positives and negatives. In addition, HEM layer contributes to release the ratio constraint on airport-background examples.
3. We develop a weight-balanced loss function to compute the average loss respectively on airport and background examples. It makes our network training focus on the hard examples and achieves a precision-recall trade-off.

The rest of this paper is organized as follows. Section 2 presents the detail of our proposed network. Section 3 analyzes the detection performance of the proposed network on a complicated and challenging dataset. Finally, Section 4 concludes this paper.

## 2. Methodology

As illustrated in Figure 1, our proposed framework consists of three main components, namely, shared convolutional neural network, region proposal network, and object detection network. These three networks are merged into a unified network to achieve an approximate joint training with hard example mining. We demonstrate that our network achieves a better training and more outstanding performance by embedding the hard example mining schemes into an end-to-end CNN training process.

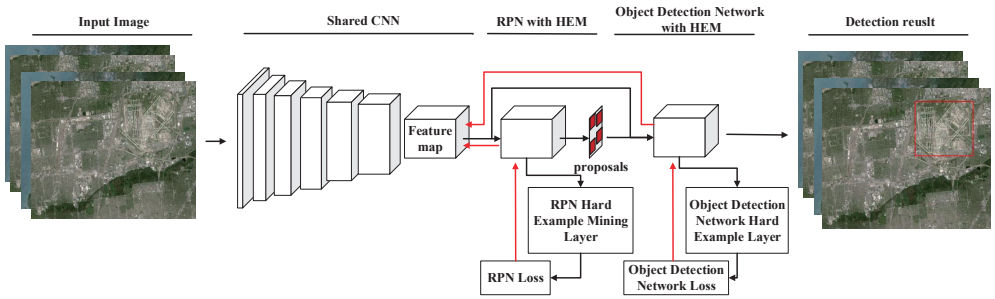



**Figure 1.** The flowchart of the proposed method. The red arrows indicate the direction of the backward propagation.

### 2.1. Shared Convolutional Neural Network

In order to describe the airport more accurately, shared convolutional neural network is constructed to extract the high-level semantic features. In addition, these features will be shared between region proposal network (RPN) and object detection network. It is conducive to improving the efficiency of the whole end-to-end network. We merge RPN and object detection network into an overall framework by using common convolutional features, rather than concatenate two independent networks.

Shared convolutional neural network is built with Zeiler and Fergus net (ZF) [31]. Since airports are typically 800–2000 m in length, 18–60 m in width [14] and RSIs used for airport detection are usually at low resolution ranging from 8 m to 60 m [1,12,25,32], we adopt the ZF net which has 5 shareable convolutional layers. In addition, the effective receptive field of the last shared convolutional layer is 139 pixels on the input image, so that an airport region is wrapped entirely.

### 2.2. Region Proposal Network with HEM

To realize the real-time end-to-end airport detection network, this paper utilizes three convolutional layers and a separate hard example mining layer to construct a region proposal network, as shown in Figure 2. Given the feature map of remote sensing image, RPN will generate a set of airport candidate regions.

In the region proposal network, the first convolutional layer has 256 kernels of size $3 \times 3$ with stride of 1 pixel. The whole feature map of size $w \times h$ is performed first by this convolutional layer to generate $w \times h \times 256$-dimensional feature tensor. In addition, each 256-dimensional features represents the context information with receptive field $3 \times 3$ in convolutional feature map which is output by the last shared convolutional layer. To harvest the airport candidate regions and their confidence, the convolutional features obtained by the first convolutional layer will be fed into the next two sibling convolutional layers, named box-regression layer and box-classification layer, which will predict the bounding box and confidence of airport candidate regions, respectively. The box-regression layer predicts the bounding box of airport candidate regions with $4 \times \alpha$ kernels of size $1 \times 1$ and the stride of 1 pixel. $\alpha$ represents the number of anchors which will be explained at next subsection. In addition, the box-classification layer evaluates the confidence of airport using $2 \times \alpha$ kernels of size $1 \times 1$ and the stride of 1 pixel. The details of two sibling convolutional layers are available in [19].

Regression and classification on bounding box both require 256-dimensional features output by first convolutional layer in RPN. In order to select hard examples from a large training set, the output of box-classification and box-regression layers will be transmitted to the hard example mining layer in the training phase, as depicted in Figure 2 with dotted line. Hard example mining layer combines the result of classification and regression on bounding box to calculate the loss on training set. In addition, the probability distribution of hard examples is constructed from the current examples losses, then the hard examples will be sampled to group as mini-batch according to that distribution. Subsequently, the later loss layer in Figure 2 will backward propagate the training error at current mini-batch and update the entire region proposal network.

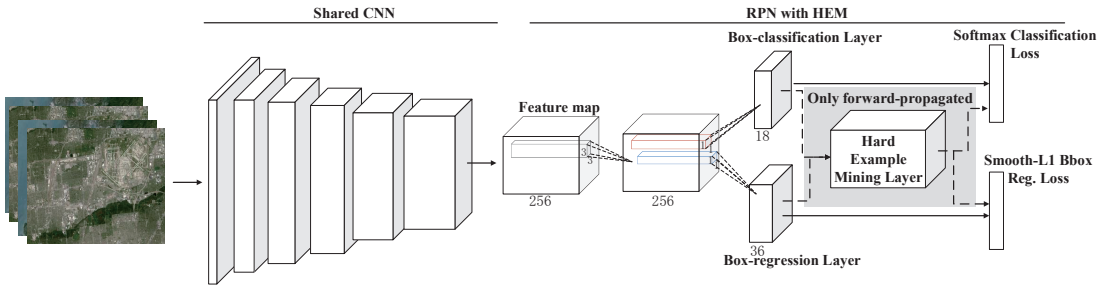



**Figure 2.** The framework of the RPN with hard example mining. The dotted line indicates that the data is only delivered in single direction along the arrow.

### 2.2.1. Anchors

Airports usually have many different orientations and scales since RSIs are taken from the upper airspace. To tackle this problem, region proposal network designs multiple scale and aspect ratio anchors at each position of feature map. The box-regression and box-classification layer calculate the calibration parameters and confidence for each anchor. After geometric calibration, the anchors are output to compute the deviation with ground-truths. In addition, the final deviation is backward propagated for updating the network parameters. Therefore, the scale and aspect ratio of anchors have a significant effect on network training.

In this paper, the selection of aspect ratio and scale is determined by training set. Before training the region proposal network, we count the aspect ratio and scale of airports respectively from the entire positive examples. As shown in Figure 3, it can be seen that the distribution of scale in the training set is mainly between $50^2$ and $270^2$ pixels. In addition, the aspect ratio of airports roughly ranges from 0.2 to 2.5. Therefore, we use three scales with box areas of $64^2$, $128^2$, $256^2$ pixels, and three aspect ratios of 0.5, 1, 2 to construct 9 anchors ($\alpha = 9$) for extracting airport candidate regions.

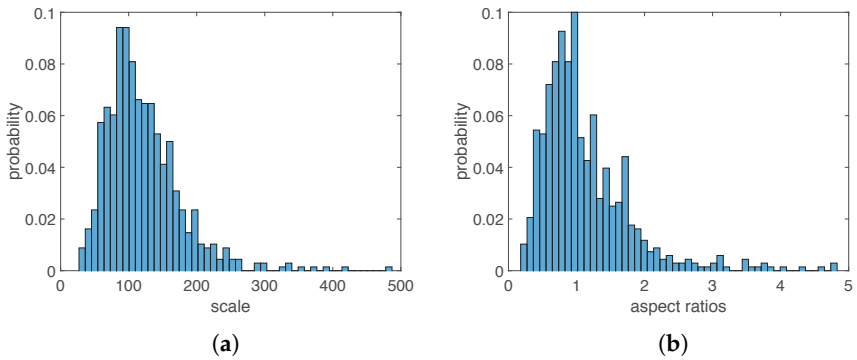



**Figure 3.** The distribution of airport regions on training set. (**a**) The distribution of scale; (**b**) the distribution of aspect ratio.

#### 2.2.2. Hard Example Mining Layer

The original region proposal network [19] balances the foreground-to-background ratio to a target of 1:1 by sampling the training examples randomly from input images. However, focusing on the airport detection in RSIs, the number of non-airport regions is usually much more than the airport regions. It is difficult to achieve the balance between positive and negative examples in each mini-batch. Such huge imbalance on training set may cause an inefficient region proposal network. In this paper, we develop a hard example mining layer that automatically selects hard examples as mini-batch in each iterative update. This layer is embedded into the RPN to inhibit spurious targets in the airport detection.

In the hard example mining layer, the intersection over union between the anchors and the ground-truths is first calculated to determine the positive anchors corresponding to the airport regions. Then, this layer receives the output of box-regression and box-classification layers to compute the loss on training set. In addition, the distribution of hard examples are constructed based on the current loss at each example. Finally, the hard examples are sampled as mini-batch to backward propagate the training error according to that distribution. In addition, a new loss function is designed to achieve precision-recall trade-off and keep the weight of airport-background class balance, as follow:

$$\begin{aligned} L_{hem}^{rpn}\left(p_i, t_i\right) &= \frac{\beta}{N_{pos}} \sum_i L_{cls}\left(p_i, p_i^*\right) \times \mathbb{1}\left[p_i^* = 1\right] \\ &+ \frac{1}{N_{neg}} \sum_i L_{cls}\left(p_i, p_i^*\right) \times \mathbb{1}\left[p_i^* = 0\right] \\ &+ \frac{\lambda}{N_{pos}} \sum_i p_i^* L_{reg}\left(t_i, t_i^*\right) \end{aligned} \tag{1}$$

Here, $L_{cls}\left(p_i, p_i^*\right)$ represents the classification log loss over airport and background. In addition, $\mathbb{1}\left[*\right]$ denotes Iverson bracket indicator function [33], which evaluates to 1 if condition in square brackets is satisfied, and 0 otherwise. So the first term and second term in this formula mean the log loss on airports and backgrounds separately. The regression loss $L_{reg}\left(t_i, t_i^*\right)$ is the smooth $L_1$ defined in [19]. $p_i$ and $p_i^*$ respectively denote the predicted and ground-truth label. Both $t_i$ and $t_i^*$ are vectors with 4 parameterized coordinates of bounding box, as mentioned in [19]. $N_{pos}$ and $N_{neg}$ are the number of airport and background examples in mini-batch, respectively. In addition, the three terms in Equation (1) are normalized by $N_{pos}$, $N_{neg}$ and two balanced parameters $\beta, \lambda$.

As can be seen from Equation (1), the first two terms compute the average loss on positive and negative examples respectively, which releases the restriction on airport-background proportion. In addition, the third term is a regression loss that enforces our network to accurately locate an airport within an image. In the HEM-CNN, we apply the hard example mining layer to select more representative and hard training example, which is much difficult to distinguish. If any hard examples were neglected, its loss would increase until it has a high probability of being sampled. At each training iteration, the most representative examples in current image would be used to optimize our network. In addition, with our designed loss function, it is not necessary to concern about the airport-background class imbalance that obstructs the network training.

### *2.3. Object Detection Network with HEM*

In our work, we utilize fast R-CNN as our object detection network and improve the performance of network by hard example mining. Object detection network is trained by approximate joint training together with region proposal network, as illustrated in Figure 4.

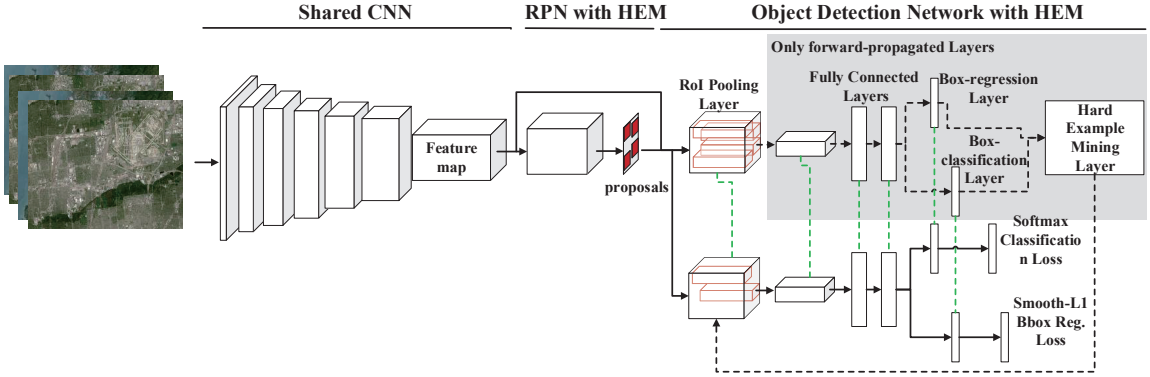



**Figure 4.** The framework of the Object Detection Network with hard example mining. The dark dotted line indicates that the data is only delivered in single direction along the arrow. In addition, loss-signals are not backward propagated along the dotted line. The network parameters are shared between the layers connected with a green dotted line.

The way of mining hard examples based on heuristic strategy in Fast R-CNN [17] is not applicable to the airport detection in RSIs. In the Fast R-CNN, the negative examples partially overlapped with the ground-truths are considered to be quite confused and easily misidentified as positives. Therefore, these examples will be regarded as hard examples among training set and screened out as essential data to optimized during network training. However, compared with natural images, the background in RSIs is much more complicated. So it is unsatisfactory to only select the negative examples partially overlapped with positives to construct mini-batch. Aiming at complex background that disturb airport detection, this paper brings the strategy of hard example mining into the object detection network. According to the current example loss, hard example mining in object detection network is no longer dependent on heuristic strategy.

In object detection network training, we score all training proposals extracted from RPN by their losses computed in object detection network. In addition, top $N_{hard}^{odn}$ proposals with highest loss will be chosen as the hard examples for training object detection network. To keep the airport-background class balance in hard examples, the loss metric $L_{hem}^{detNet}$ in object detection network is similar to $L_{hem}^{rpn}(p_i, t_i)$, as follows.

$$\begin{aligned} L_{hem}^{detNet}(p_i, t_i) = & \frac{\beta}{N_{pos}} \sum_i L_{cls}(p_i, p_i^*) \times \mathbb{1}[p_i^* = 1] \\ & + \frac{1}{N_{neg}} \sum_i L_{cls}(p_i, p_i^*) \times \mathbb{1}[p_i^* = 0] \\ & + \frac{\lambda}{N_{pos}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \end{aligned} \tag{2}$$

## 2.4. Approximate Joint Training

Due to a large amount of labeled data frequently required to train a CNN model, it is insufficient to learn so many network parameters using limited airport examples in RSIs [25]. However, the optical remote sensing images has close similarity with natural images at low-level and middle-level visual features. This paper follows the work in [25] that transfers the convolutional neural network trained from a large number of natural images to airport detection in RSIs, and applies the labeled airport dataset to fine-tune the pre-training network. To reduce over-fitting during fine-tuning phase, we employ data augmentation on the airport training set. The airport training set is augmented to obtain 48× training examples by horizontal, vertical reflection and rotating with different orientations ranged from 30° to 330°, with step 30°.

In this paper, the approximate joint training method is used to train the shared convolutional neural network, RPN and object detection network in the unified framework, as shown in Figure 1. In each stochastic gradient descent (SGD) iteration, the forward propagation first transmits the input image into the shared convolutional neural network to extract the feature map. Then, the feature map is employed to generate the region proposals in region proposal network. Finally, the extracted region proposals will be used to train the object detection network. The backward propagation takes place as usual for RPN and object detection networks [19]. Only for the shared convolutional neural network, the backward propagated signals are dependent on both the RPN loss and object detection network loss.

## 3. Results and Discussion

### *3.1. Dataset and Parameter Setting*

To prove the performance of our method, we extend a complicated and challenging optical remote sensing dataset [1] with 244 remote sensing images acquired from Google Earth, including 148 different airports at different spatial resolutions. These remote sensing data contain airport images in different regions with cluttered background, such as mountains, islands, cities, farmland, desert, hills and so on. There are various viewpoints and illumination intensities in our dataset. The size of images is 500 $\times$ 600 pixels and spatial resolution ranges from 8 m to 60 m. There are 974 images in total and ground-truth data are labeled manually. In our experiments, we implement a 5-fold cross validation for the training and testing to increase the confidence in the numerical results. In addition, there are 20% images using for validation in training set. It results in three subsets: a training set containing 623 images, a validation set containing 156 images, and a test set containing 195 images. Based on the dataset partition, the training set is augmented to obtain 48$\times$ training images.

In our work, the HEM-CNN is initialized by a pre-training model learned from natural images [31], as a proper practice [25]. We fine-tune all layers of the ZF net for 40 k iterations with a momentum of 0.9, a weight decay of 0.0005, and a learning rate of 0.001. The parameters of $\lambda$ and $\beta$ are chosen as 1 and 0.25, with respectively. The validation results are presented in Table 1. The balanced parameter $\lambda$ is set to 1 to make classification and regression terms equally weighted. In addition, we set $\beta = 0.25$ to inhibit the false positives for complicate background in RSIs. We configure mini-batch size to 256 for training region proposal network. To train object detection network, all proposals generated from RPN are first sorted based on their score. Then we pick up the top 12,000 proposals for non-maximum suppression, and finally select the top 2000 proposals with high score to train the object detection network. The mini-batch size in object detection network is set to 128 ($N_{hard}^{odn} = 128$).

The experiments were carried on a Windows server with an Intel(R) Xeon(R) CPU E5-2620 v2 @2.10GHz and a NVIDIA Tesla K40c GPUs with 12 GB onboard memory.

**Table 1.** Network performance on validation set according to different $\beta$ and $\lambda$.

| $\beta$ | **0.1** | **0.1** | **0.1** | **0.25** | **0.25** | **0.25** | **0.4** | **0.4** | **0.4** |
|---|---|---|---|---|---|---|---|---|---|
| $\lambda$ | **0.5** | **1** | **1.5** | **0.5** | **1** | **1.5** | **0.5** | **1** | **1.5** |
| Precision | 0.979 | **0.9949** | 0.9803 | 0.98 | 0.9797 | 0.975 | 0.9753 | 0.9481 | 0.9209 |
| Recall | 0.8947 | 0.9378 | 0.9433 | 0.9378 | **0.9554** | 0.933 | 0.9339 | 0.9481 | 0.934 |
| F1-Measure | 0.935 | 0.9655 | 0.9614 | 0.9584 | **0.9674** | 0.9535 | 0.9542 | 0.9481 | 0.9274 |
| AP | 0.951 | 0.9602 | 0.9552 | 0.9577 | **0.9714** | 0.9627 | 0.9548 | 0.9693 | 0.9689 |

### 3.2. Evaluation Metrics

In these experiments, we employ the precision-recall (PR) curve, average precision (AP) [13], IoU-Recall (IR) curve [34], and F1-measure [28] to quantitatively evaluate the performance of our HEM-CNN method. They are standard and widely used metrics in many target detection works, such as those in [1,11,13,24,26,28,35].

The detection is considered to be correct when the intersection over union (IoU) between detection box and ground-truth bounding box exceeds 40%, otherwise the detection box is assumed as a false positive. IoU is defined as following formula:

$$IoU = \frac{area\left(B_{det} \bigcap B_{gt}\right)}{area\left(B_{det} \bigcup B_{gt}\right)} \times 100\% \tag{3}$$

Here, $area\left(B_{det} \bigcap B_{gt}\right)$ indicates the intersection area between detection box and ground-truth bounding box, and $area\left(B_{det} \bigcup B_{gt}\right)$ indicates their union.

### 3.3. Verification of HEM and Weight-Balanced Loss Function

To discuss the airport-background class imbalance in RSIs, we made statistics about positive and negative examples on the training set. As depicted in Table 2, we can find that the training set contains airport and background examples with a ratio of 1:16. However, the RPN [19] and Fast R-CNN [17] require training set composed of positive and negative examples with a ratio of 1:1 and 1:3, with respectively. It proves that there exists a large imbalance on airport-background class examples for training airport detector. Therefore, it is necessary to select more representative examples, for keeping airport-background examples balance and avoiding the degeneration of network training.

**Table 2.** The statistics of examples on training set.

| | **Positives** | **Negatives** |
|---|---|---|
| Examples number | 25,712 | 409,374 |
| Hard examples number $(loss > 0)$ | 9030 | 19,573 |
| Average Loss | 5.791 | −0.897 |

Given the training set, the goal of network training is to minimize the loss function by updating layer parameters. So we exploit a loss oriented strategy and further analyze the loss proportion on positive and negative examples. As shown in Figure 5 and Table 2, it can be concluded that there are much more negative examples with lower loss than positives. In this paper, we define the difficulty of example by its loss, which means the example with higher loss is more difficult to distinguish by neural network. In addition, top mini-batch-size examples with highest losses will be chosen as hard examples for network training. Otherwise examples are assumed as easy examples. The hard and easy examples are shown in Figure 5. Comparing hard ones with easy ones in images, we can found following characteristics:

1. In contrast with hard examples, there are much more easy examples in RSIs.
2. Easy examples mainly concentrate on the background areas in RSIs, that is to say, negatives have more easy examples than positives. So it is feasible to employ hard example mining to handle the airport-background class imbalance.

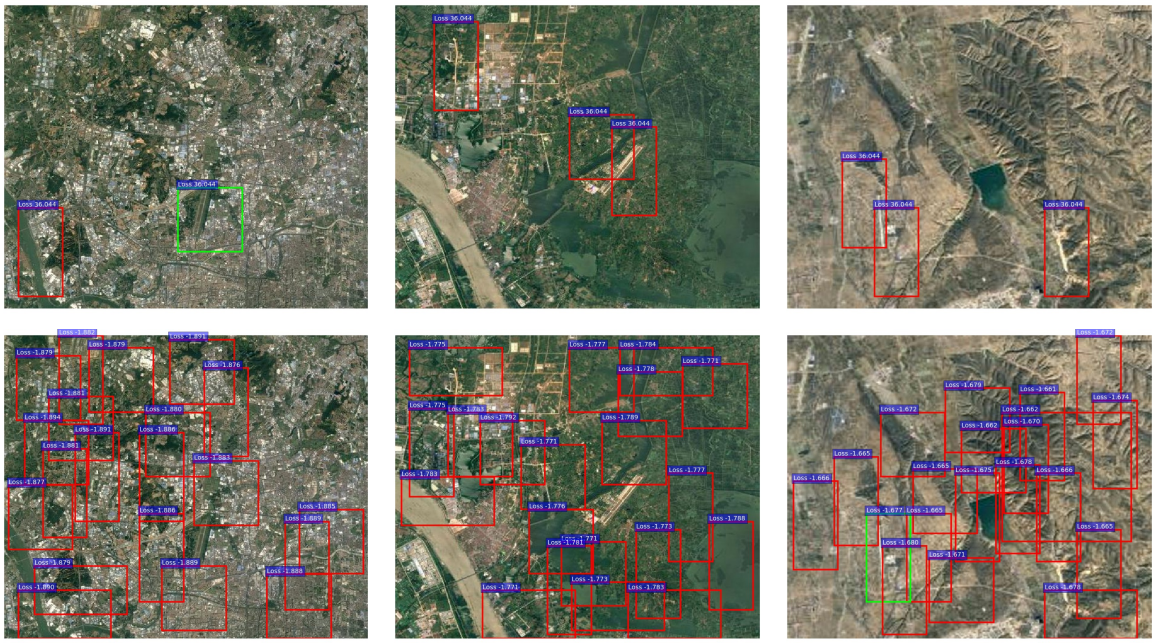


**Figure 5.** Hard and easy examples. The first row lists hard examples. The second row denotes easy examples. The positive and negative examples are indicated by green and red rectangles, respectively.

Besides the mathematical definition of hard example above, we made a further study which probes into the physical significance on hard example mining. For the task of airport detection, we found out that most of hard examples formed into 6 categories including highway, salient roadway, salient mountain ridge, riverbank, coastline, and regions partially overlapped with airport. It can be seen from Figure 6a–c that highway, coastline and riverbank are always regarded as hard examples, since they are similar with airport runway. Regions in Figure 6d,e are much more salient than other areas and have some line features, which are commonly used in airport region extraction [13]. As for regions in Figure 6f, it is hard for network to recognize their classes, due to the region overlapped with airport. Faster R-CNN [19] also utilizes this heuristic region-overlap strategy to select hard examples. From the above analysis on hard examples, it is reasonable to use our HEM strategy to select representative examples for network training.

In order to evaluate the effectiveness of the hard example mining and weight-balanced loss function, we make comparison on Faster R-CNN [19], Faster R-CNN with weight-balanced loss, HEM without weight-balanced loss and HEM with weight-balanced loss. The experimental results are shown in Figure 7 and Table 3, which prove that training network with HEM could promote the performance of final network. Despite the number of negative examples is greater than positives, our designed loss function also makes network optimization treat positives and negatives equally. Our HEM-CNN method focuses more on hard examples with high loss, which improves the efficiency of network training.

**Table 3.** Comparison of HEM and new loss function.

| Method | Precision | Recall | F1-Measure | AP |
|---|---|---|---|---|
| Faster R-CNN [19] | 0.8881 | 0.9386 | 0.9127 | 0.9462 |
| Faster R-CNN with weight-balanced loss | 0.9299 | 0.9387 | 0.9343 | 0.9472 |
| HEM without weight-balanced loss | **0.9863** | 0.9283 | 0.9564 | 0.9553 |
| HEM with weight-balanced loss | 0.9671 | **0.9465** | **0.9567** | **0.9702** |

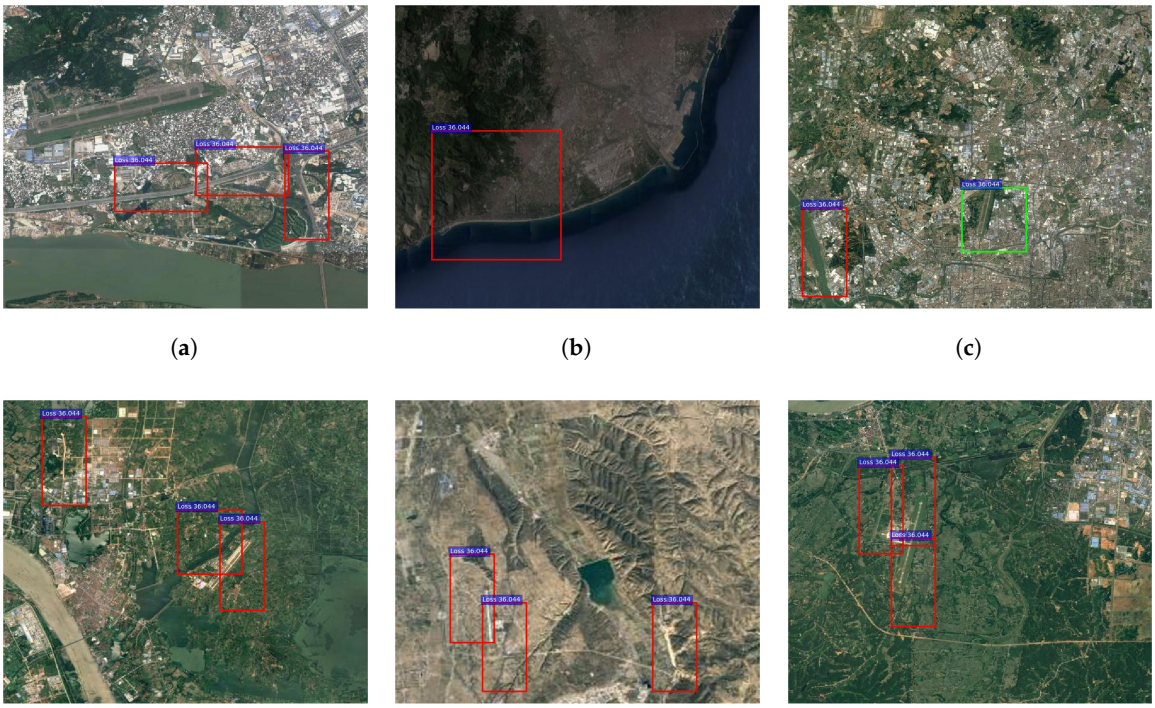


**Figure 6.** The 6-class hard examples. The positive and negative examples are indicated by green and red rectangles, respectively. (**a**) highway; (**b**) coastline; (**c**) riverbank; (**d**) salient roadway; (**e**) salient mountain ridge; (**f**) regions overlapped with airport .

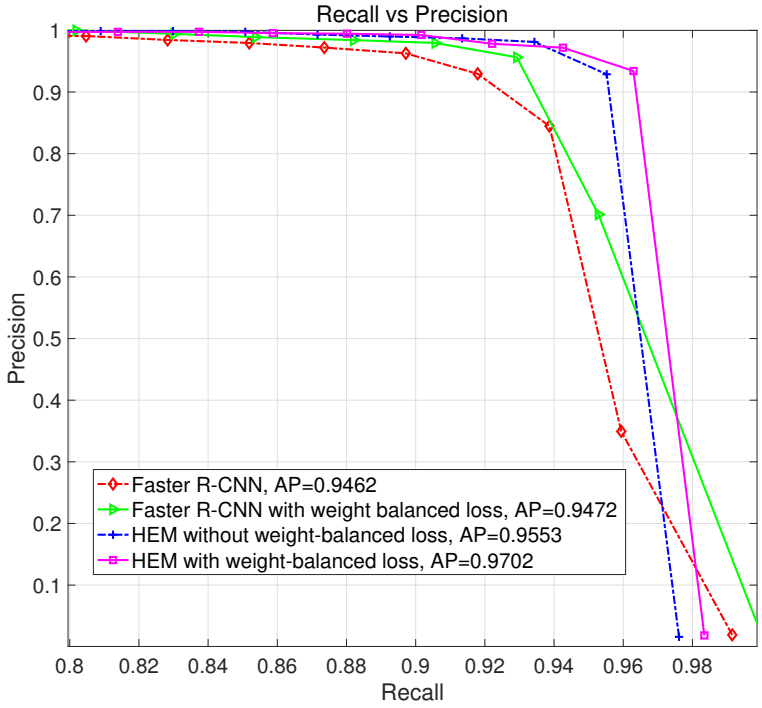



**Figure 7.** Comparison of HEM and new loss function.

### *3.4. Performance on Region Proposal Method*

To validate the capability of our region proposal network, we compare our RPN with different proposal extraction methods, including Saliency and CRF [13], RPN [19], selective search [36] commonly used in object detection [24,26]. The Saliency and CRF method extracts airport candidate regions, based on the contrast and straight-line of airport runways. For selective search method, we select top 1500 proposals as the airport candidate regions according to their priorities [26]. In RPN [19] and our RPN with HEM, top 300 proposals with highest scores are chosen as the airport candidate regions. To quantitatively evaluate proposal extraction method, the IoU-Recall curve is used to analyze airports recall in testing set, since missed airport regions cannot be recovered in subsequent object detection stage.

Figure 8 illustrates the performance of different proposal extraction methods. Considering the trade-off between localization accuracy and recall, our network achieves the best performance than other methods. Although the selective search method has a higher recall at low IoU, the candidate regions simultaneously contain much more useless background information, which will disturb the correct recognition on airport regions in airport identification. When IoU is higher than 0.2, our proposal method has the highest recall than other methods, which confirms that hard example mining can improve the recall of candidate region extraction. Moreover, the region extraction method with excellent performance should not only achieve a good recall, but also have a low false positive rate. Therefore we applied the PR curve to make a further comparison on this two region proposal networks and analyzed the effect of the hard example mining in region proposal network. It can be found from Figure 9b that RPN with HEM has a higher precision and recall than the original RPN [19], since it is beneficial for network training to use hard examples which are always too confused and difficult to recognized by traditional detector. Bringing hard example mining into RPN achieves a better recall, and extremely suppresses the false positives in the proposal extraction stage.

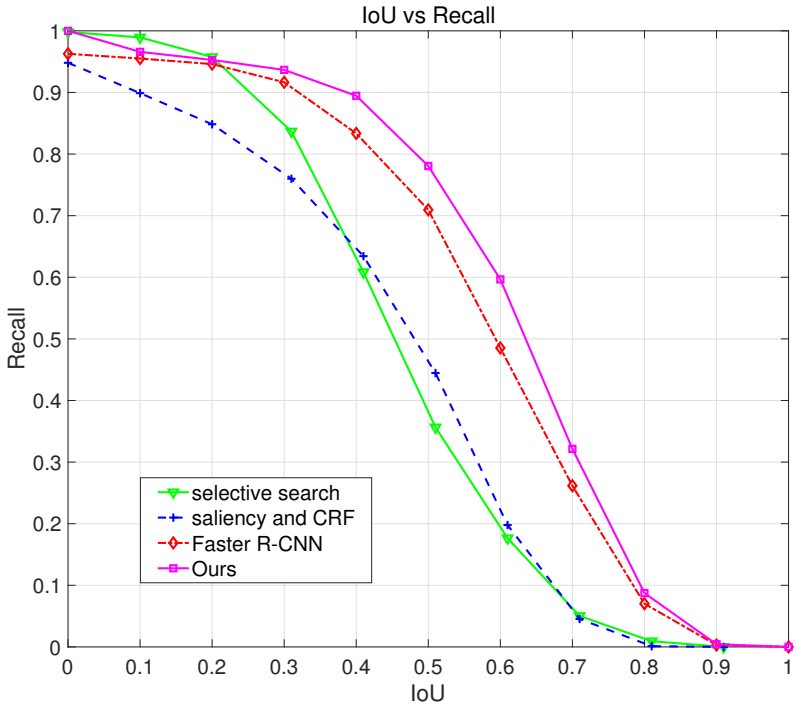
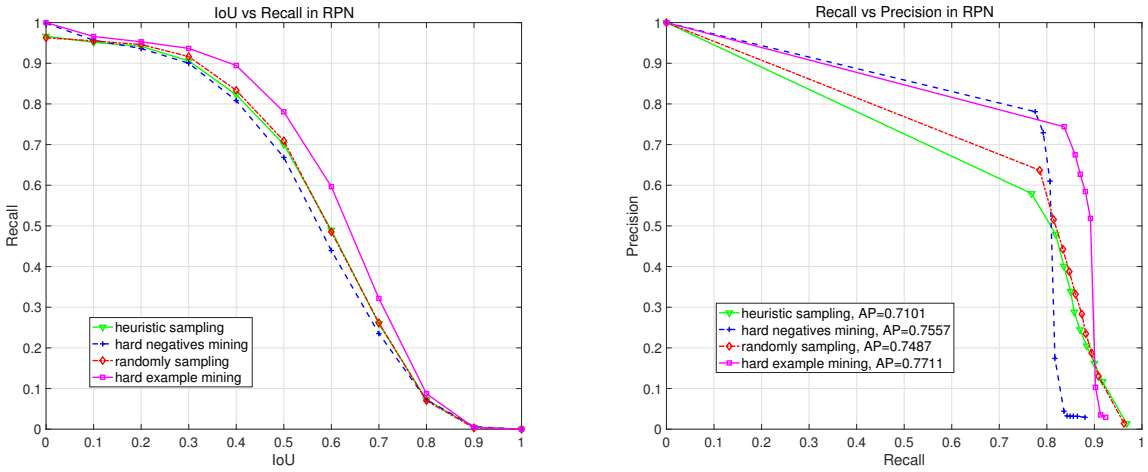



**Figure 8.** Comparison of the different detection proposal methods.




(**a**) (**b**)

**Figure 9.** Comparison of the different sampling methods on region proposal. (**a**) IoU-Recall Curve in RPN; (**b**) Precision-Recall curve in RPN.

### *3.5. Comparison on Different Sampling Methods*

#### 3.5.1. Overview Sampling Methods

We also implement a experiment to verify which sampling method is the most excellent for network training. The sampling methods in this experiment are including random sampling, heuristic sampling [17], hard negatives mining [23,28] and our hard examples mining. These three sampling methods except ours all sample airport and background examples with a ratio of 1:1 in RPN and 1:3 in object detection network, respectively. In the experiment, we configure the mini-batch size to 256 in RPN and 128 in object detection network. If there are a few airport examples in an image, we will pad the mini-batch with backgrounds. The random sampling method samples airport and background examples in an image based on the uniform distribution. Heuristic sampling following [17], selects background examples with some overlap with ground-truths, since these examples are more likely to be confused. Hard negatives mining method activated by the work in [23], computes the loss of all background examples in an image and sorts these examples by their losses, then takes examples for which the current network performs worst. Hard example mining layer proposed in this paper, sorts airport and background examples together and selects examples with higher loss as mini-batch.

#### 3.5.2. Evaluating Sampling Method on Region Proposal

Figure 9 demonstrates the quantitative comparison results of four different sampling methods, measured by IoU-Recall curve and PR curve, respectively. The experimental results show that it is capable to greatly improve the recall and precision of the region proposal network by introducing the hard example mining layer into the RPN. Compared to other sampling methods, only HEM simultaneously selects airport and background examples by their losses which consists of two parts: the error of classification and localization, as shown in Equation (1). Therefore, hard example mining not only considers the correct classification, but also focuses on the accuracy of localization. The heuristic sampling method selects hard background examples overlapped with ground-truths. However, random sampling collects mini-batch examples at any location of whole image. It can grasp all kinds of examples in complicate RSI. So random sampling has a better precision than heuristic sampling method. The hard negatives mining method has a worse recall than other methods, and the reason may be the huge loss gap between the background and airport examples. Even if the mini-batch contains airport and background examples, network training mainly concentrates on the correct classification

on these hard backgrounds. In addition, the ratio between airport and background examples also limits the effective mining on hard background examples, resulting in the missing detection on airports.

#### 3.5.3. Evaluating Sampling Method on Final Detection

After comparing different sampling methods on region proposal network, we further evaluate the effect of sampling methods on the final detection. As shown in Figure 10, our hard example mining method achieves the best performance. Although negatives mining obtains a higher precision, it also misses more airport targets as depicted in Table 4. Heuristic sampling has a high recall, but its precision is the worst in the four sampling methods. This adequately shows that heuristic sampling ignores some infrequent, but important, difficult background regions without overlap with ground-truths. However, hard example mining is directly dependent on the example loss at network forward propagation, rather than a spatial priori information used in heuristic sampling. So hard background examples with higher loss will attract more attention, which greatly inhibit the spurious airport regions and improve the detection accuracy. Based on the optimal sampling method determined above, our network adopts hard example mining as our sampling method for network training.

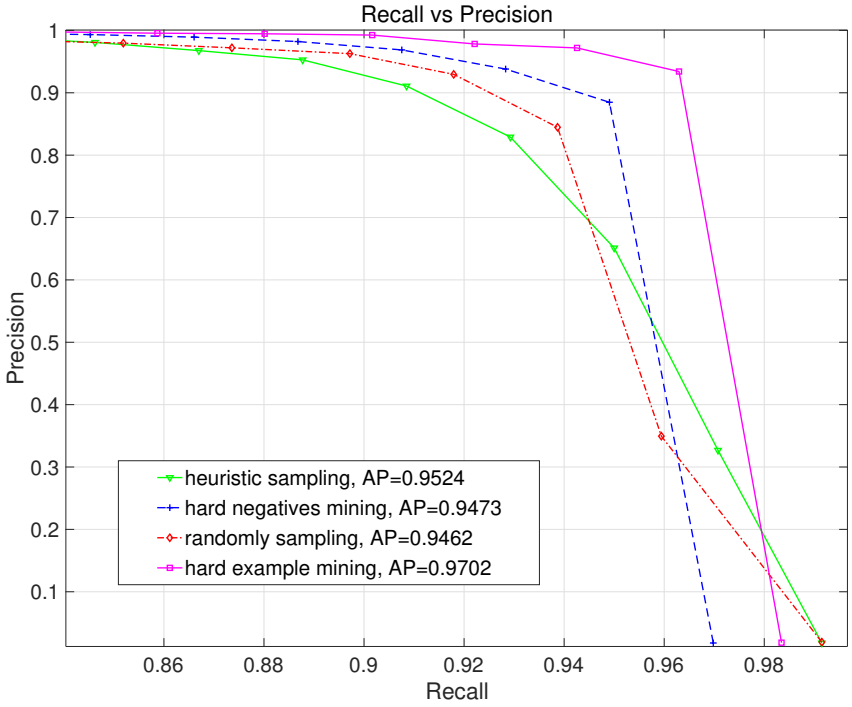



**Figure 10.** Comparison of the different sampling methods on final detection.

**Table 4.** Comparison of the different sampling methods on final detection.

| Sampling Method | Random Sampling | Heuristic Sampling [17] | Hard Negatives Mining [23,28] | Hard Example Mining |
|---|---|---|---|---|
| Precision | 0.8881 | 0.7478 | **0.9773** | 0.9671 |
| Recall | 0.9386 | 0.9443 | 0.8971 | **0.9465** |
| F1-Measure | 0.9127 | 0.8347 | 0.9355 | **0.9567** |
| AP | 0.9462 | 0.9524 | 0.9473 | **0.9702** |

Comparing the results in Figures 9b and 10, we can find that object detection network improves the recall and precision in final detection. This shows that the region proposal network with relatively low recall is probably due to the poor localization of RPN. The candidate regions extracted from RPN not only cover the airport area, but also contain the extra background area, so the IoU between proposals and ground-truths is relatively low. Object detection network enhances localization accuracy using bounding-box regression and eliminates the false positives generated in proposal extraction stage. Therefore the object detection network has an irreplaceable role in our HEM-CNN method.

### *3.6. Comparison on State-of-the-Art Airport Detection Methods*

To quantitatively evaluate our network, we compared it with three traditional state-of-the-art methods and two CNN methods, in which the BU and TD saliency [12], the HRL [1], the saliency and CRF [13], CNN with USB-BBR [26] and the Faster R-CNN model [19] are employed, respectively. For a fair comparison, we applied the same training set and testing set for HEM-CNN and other comparison methods.

Table 5, Figures 11 and 12 show the comparison results of different methods, measured by Precision, Recall, AP values, F1-Measure, and PR curve, respectively. A similar conclusion can be made for the experiment results, in that our proposed method outperform all other methods and the average computing time is about 0.12 seconds per image. The methods in [1,12,13] combine saliency with the line characteristics of the airport runways to detect airport regions. However, it is inefficient to identify the airport regions in complex and cluttered environment with multiple spatial resolutions. As shown in Figures 12–14, there are a lot of false positives in traditional airport detection results. Because traditional airport detection methods can be disturbed by targets with line features, such as roads, riverbank, and regular distribution of farmland. Compared with the traditional manually designed low-level features and saliency features, the high-level features extracted by CNN have a better expressive power on airports. It can be seen from Table 5 that CNN methods are more accurate than traditional airport detection methods. However, CNN with USB-BBR [26] has a worse recall than traditional methods, since the selective search method could not extract plentiful airport candidate regions. Therefore, it is necessary to adopt an end-to-end network structure, which applies high-level features sufficiently for candidate region extraction and airport identification.

It can be seen from Table 5 that Faster R-CNN [19] achieves a better precision and recall than CNN with USB-BBR [26] and traditional airport detection methods [1,12,13]. However, as shown in Figures 12–14, Faster R-CNN [19] still misses a large amount of airport regions with small size, due to the example with small size has less available information in RSI. These examples with small size are much harder and should be taken into more consideration by the network. Faster R-CNN [19] adopts a random sampling method to obtain mini-batch for updating the network. Random sampling equally selects examples regardless of their size or difficulty. It results in ignoring the correct classification on hard examples with small size, since easy examples can overwhelm training. Therefore, airport targets with small size are miss detected in the RSIs. By contrast, our HEM-CNN method is more robust than traditional airport detection methods under different illuminations and viewpoints. Meanwhile, comparing with the Faster R-CNN method, the embedding of hard example mining and new loss function enhances the network adaptability in complicated imaging conditions.

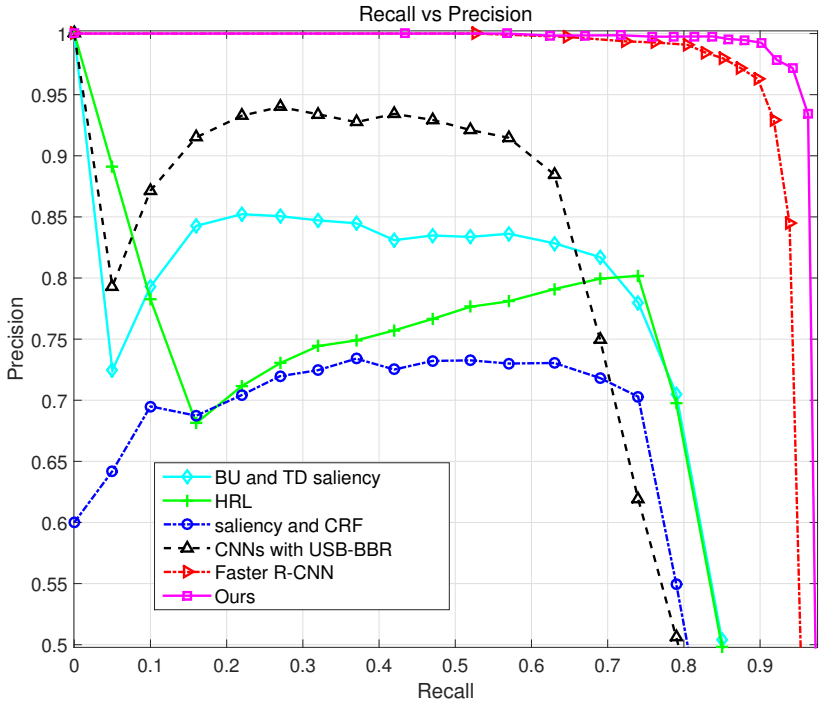



**Figure 11.** Comparison of the different airport detection methods.

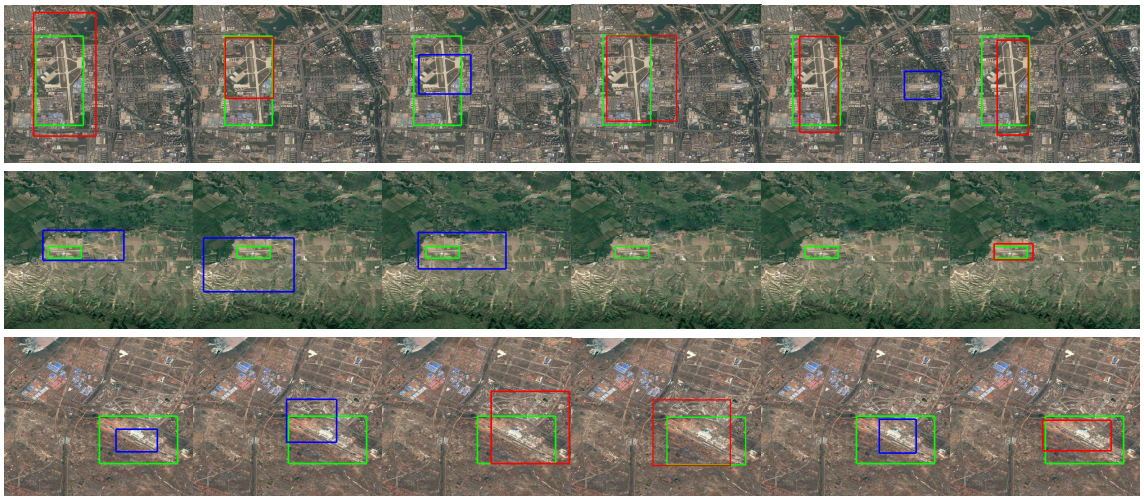
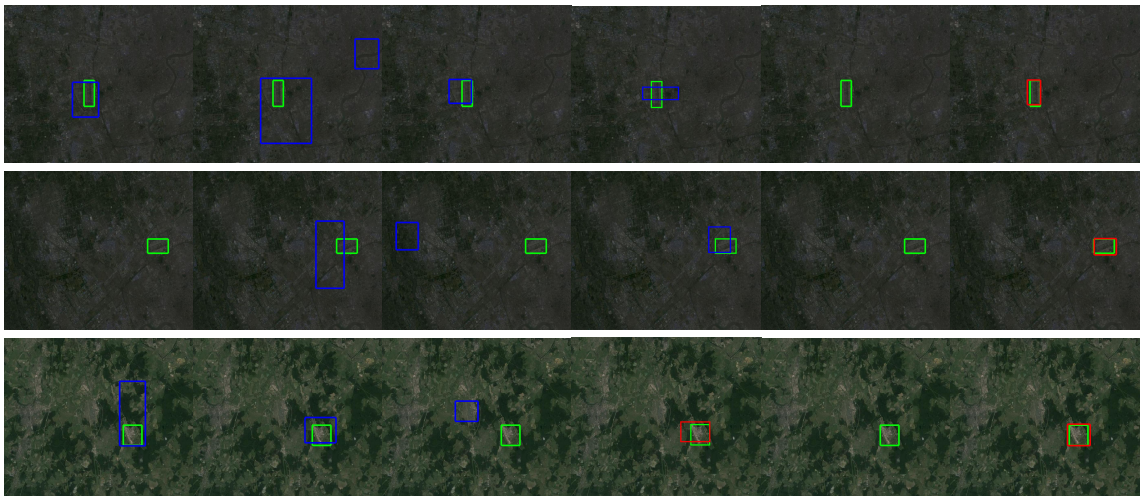
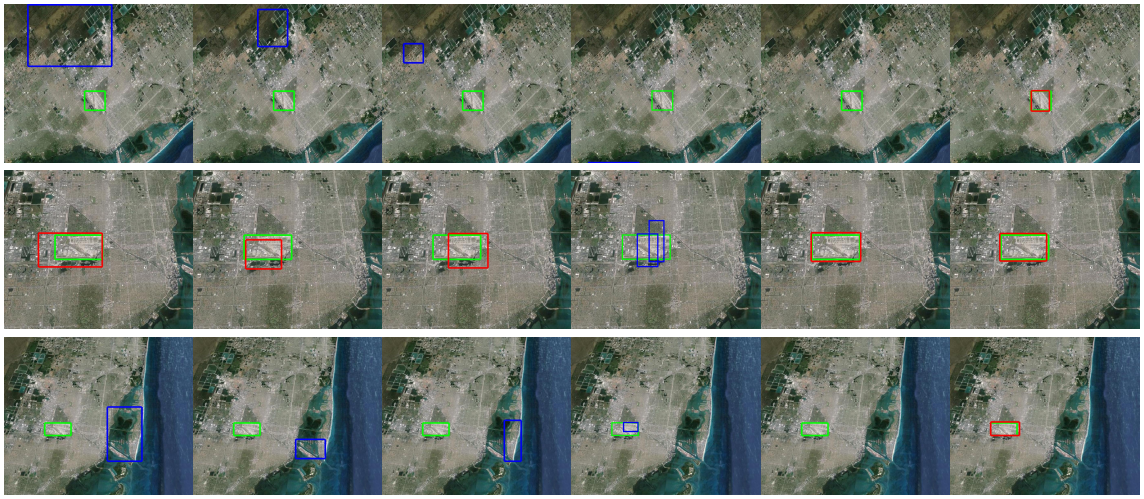



**Figure 12.** Detection results of different airport detection methods. The true positives, false positives, and ground-truths are indicated by red, blue and green rectangles, respectively. (**a**) BU and TD saliency [12]. (**b**) Saliency and CRF [13]. (**c**) HRL [1]. (**d**) CNN with USB-BBR [26]. (**e**) Faster R-CNN [19]. (**f**) our network.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

**Figure 13.** Detection results at different illuminations. The true positives, false positives, and ground-truths are indicated by red, blue and green rectangles, respectively. (**a**) BU and TD saliency [12]. (**b**) Saliency and CRF [13]. (**c**) HRL [1]. (**d**) CNN with USB-BBR [26]. (**e**) Faster R-CNN [19]. (**f**) our network.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

**Figure 14.** Detection results at different viewpoints. The true positives, false positives, and ground-truths are indicated by red, blue and green rectangles, respectively. (**a**) BU and TD saliency [12]. (**b**) Saliency and CRF [13]. (**c**) HRL [1]. (**d**) CNN with USB-BBR [26]. (**e**) Faster R-CNN [19]. (**f**) our network.

**Table 5.** Comparison of the different airport detection methods.

| Detection Method | BU and TD Saliency [12] | HRL [1] | Saliency and CRF [13] | CNN with USB-BBR [26] | Faster R-CNN [19] | Ours |
|---|---|---|---|---|---|---|
| Precision | 0.8096 | 0.8077 | 0.7145 | 0.8411 | 0.8881 | **0.9671** |
| Recall | 0.7188 | 0.7283 | 0.716 | 0.6575 | 0.9386 | **0.9465** |
| F1-Measure | 0.7615 | 0.7659 | 0.7152 | 0.7381 | 0.9127 | **0.9567** |
| AP | 0.6046 | 0.5845 | 0.5549 | 0.6427 | 0.9462 | **0.9702** |
| Average time (s) | 2.12 | 3.52 | 1.97 | 4.90 | 0.13 | **0.12** |

### *3.7. Performance Analysis on Illumination and Resolution*

#### 3.7.1. Illumination

To analyse the effect of illumination on the proposed method, we divide the testing set into two different illumination subsets. Table 6 shows the detection results of the proposed method under different illuminations. We found that higher precision and recall are achieved in sufficient illumination. Although the proposed method has a performance decay in deficient illumination, the difference on precision and recall is less than 1% between sufficient and deficient illuminations. Therefore, it can be concluded from this result that our proposed method is robust to the different illuminations.

**Table 6.** The effect of illumination on our proposed method.

| Illumination | Sufficiency | Deficiency |
|---|---|---|
| Precision | 0.9701 | 0.9613 |
| Recall | 0.9498 | 0.9405 |

#### 3.7.2. Resolution

According to the spatial resolution of images, we divide testing set into four levels, where level 1 is 36 to 60 m/pixel, level 2 is 18 to 36 m/pixel, level 3 is 9 to 18 m/pixel, and level 4 is 9 m/pixel or less. We summarize the detection results as shown in Table 7. It can be seen that the proposed method has the lowest recall with 92.84% at level 1, and gets the worst precision at level 4. The reason why precision has a decay in the higher resolution is that our network extracts a few regions which are part of airports. In addition, these regions are assumed as false positives due to the strict metric IoU ($\geq$40%) used in our experiments. At a lower resolution, although the HEM can reduce miss detections by focusing more on regions with small size, the limited spatial information still weakens the localization accuracy, and results in misdetections.

**Table 7.** The effect of resolution on our proposed method.

| Resolution | Level 1 | Level 2 | Level 3 | Level 4 |
|---|---|---|---|---|
| Precision | 1 | 0.9881 | 0.9619 | 0.8913 |
| Recall | 0.9284 | 0.9513 | 0.9538 | 0.9441 |

## 4. Conclusions

In this paper, we have proposed an end-to-end HEM-CNN framework to tackle the airport-background class imbalance in CNN training. The end-to-end structure of HEM-CNN would release the constraint on image size and accelerate the network training. In addition, the cascade design of RPN and object detection network reduces false positives tremendously. Our framework contains a new designed hard example mining layer, which selects more representative examples for network training. To achieve precision-recall trade-off, we train HEM-CNN framework by a weight-balanced loss function. It is efficient to train our network by hard example mining and weight-balanced loss function. The experiments demonstrated that the high-level features extracted by CNN are more robust than manually designed features for airport detection. RPN in HEM-CNN achieves high-quality region proposals by this high-level features. From the experimental results, we find that our HEM-CNN could obtain a outstanding performance on airport detection in complicated environment. In the future work, we will further study the impact of hard example mining on network training, and enhance the detection accuracy of RPN so that it can replace the object detection network to efficiently accomplish the airport detection task.

**Acknowledgments:** This work was supported in part by the National Key Research and Development Program of China (2016YFB0501300, 2016YFB0501302), the National Natural Science Foundation of China (Grant Nos. 61501009, 61371134 and 61071137) and Aerospace Science and Technology Innovation Fund of CASC.

**Author Contributions:** Bowen Cai, Zhiguo Jiang, Haopeng Zhang, Danpei Zhao, and Yuan Yao conceived of this study. Bowen Cai performed the experiments and wrote the paper. Zhiguo Jiang and Haopeng Zhang analyzed the data and wrote the paper. Danpei Zhao and Yuan Yao implemented the experimental validation and commented on the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Zhao, D.; Ma, Y.; Jiang, Z.; Shi, Z. Multiresolution Airport Detection via Hierarchical Reinforcement Learning Saliency Model. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2017**, *10*, 2855–2866.
2. Budak, Ü.; Halıcı, U.; Şengür, A.; Karabatak, M.; Xiao, Y. Efficient Airport Detection Using Line Segment Detector and Fisher Vector Representation. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 1079–1083.
3. Han, X.; Zhong, Y.; Zhang, L. An Efficient and Robust Integrated Geospatial Object Detection Framework for High Spatial Resolution Remote Sensing Imagery. *Remote Sens.* **2017**, *9*, doi:10.3390/rs9070666.
4. Han, X.; Zhong, Y.; Cao, L.; Zhang, L. Pre-Trained AlexNet Architecture with Pyramid Pooling and Supervision for High Spatial Resolution Remote Sensing Image Scene Classification. *Remote Sens.* **2017**, *9*, 848, doi:10.3390/rs9080848.
5. Pan, B.; Shi, Z.; Xu, X. R-VCANet: A New Deep-Learning-Based Hyperspectral Image Classification Method. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2017**, *10*, 1975–1986.
6. Panboonyuen, T.; Jitkajornwanich, K.; Lawawirojwong, S.; Srestasathiern, P.; Vateekul, P. Road Segmentation of Remotely-Sensed Images Using Deep Convolutional Neural Networks with Landscape Metrics and Conditional Random Fields. *Remote Sens.* **2017**, *9*, 680, doi:10.3390/rs9070680.
7. Pan, B.; Shi, Z.; An, Z.; Jiang, Z.; Ma, Y. A Novel Spectral-Unmixing-Based Green Algae Area Estimation Method for GOCI Data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2017**, *10*, 437–449.
8. Kou, Z.; Shi, Z.; Liu, L. Airport detection based on line segment detector. In Proceedings of the IEEE 2012 International Conference on Computer Vision in Remote Sensing (CVRS), Xiamen, China, 16–19 December 2012; pp. 72–77.
9. Aggarwal, N.; Karl, W.C. Line detection in images through regularized Hough transform. *IEEE Trans. Image Process.* **2006**, *15*, 582–591.
10. Xiong, W.; Zhong, J.; Zhou, Y. Automatic recognition of airfield runways based on Radon transform and hypothesis testing in SAR images. In Proceedings of the IEEE 2012 5th Global Symposium on Millimeter Waves (GSMM), Harbin, China, 27–30 May 2012; pp. 462–465.
11. Zhang, L.; Zhang, Y. Airport Detection and Aircraft Recognition Based on Two-Layer Saliency Model in High Spatial Resolution Remote-Sensing Images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, doi:10.1109/JSTARS.2016.2620900.
12. Zhu, D.; Wang, B.; Zhang, L. Airport Target Detection in Remote Sensing Images: A New Method Based on Two-Way Saliency. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 1096–1100.
13. Yao, X.; Han, J.; Guo, L.; Bu, S.; Liu, Z. A coarse-to-fine model for airport detection from remote sensing images using target-oriented visual saliency and CRF. *Neurocomputing* **2015**, *164*, 162–172.
14. Tao, C.; Tan, Y.; Cai, H.; Tian, J. Airport detection from large IKONOS images using clustered SIFT keypoints and region information. *IEEE Geosci. Remote Sens. Lett.* **2011**, *8*, 128–132.
15. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110.
16. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
17. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
18. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Victoria, BC, Canada, 1–3 June 2016; pp. 779–788.

19. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the 28th International Conference on Computer Vision, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
20. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the European conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Cham, Switzerland, 2016; pp. 21–37.
21. Li, W.; Fu, H.; Yu, L.; Cracknell, A. Deep Learning Based Oil Palm Tree Detection and Counting for High-Resolution Remote Sensing Images. *Remote Sens.* **2017**, *9*, 22, doi:10.3390/rs9010022.
22. Bejiga, M.B.; Zeggada, A.; Nouffidj, A.; Melgani, F. A Convolutional Neural Network Approach for Assisting Avalanche Search and Rescue Operations with UAV Imagery. *Remote Sens.* **2017**, *9*, 100, doi:10.3390/rs9020100.
23. Zhang, F.; Du, B.; Zhang, L.; Xu, M. Weakly Supervised Learning Based on Coupled Convolutional Neural Networks for Aircraft Detection. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 5553–5563.
24. Cheng, G.; Zhou, P.; Han, J. Learning Rotation-Invariant Convolutional Neural Networks for Object Detection in VHR Optical Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 7405–7415.
25. Zhang, P.; Niu, X.; Dou, Y.; Xia, F. Airport detection from remote sensing images using transferable convolutional neural networks. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016; pp. 2590–2595.
26. Long, Y.; Gong, Y.; Xiao, Z.; Liu, Q. Accurate Object Localization in Remote Sensing Images Based on Convolutional Neural Networks. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 2486–2498.
27. Yao, Y.; Jiang, Z.; Zhang, H.; Zhao, D.; Cai, B. Ship detection in optical remote sensing images based on deep convolutional neural networks. *J. Appl. Remote Sens.* **2017**, *11*, doi:10.1117/1.JRS.11.042611.
28. Cheng, G.; Han, J.; Guo, L.; Qian, X.; Zhou, P.; Yao, X.; Hu, X. Object detection in remote sensing imagery using a discriminatively trained mixture model. *ISPRS J. Photogramm. Remote Sens.* **2013**, *85*, 32–43.
29. Shrivastava, A.; Gupta, A.; Girshick, R. Training Region-Based Object Detectors with Online Hard Example Mining. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 761–769.
30. Wang, X.; Gupta, A. Unsupervised Learning of Visual Representations Using Videos. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; pp. 2794–2802.
31. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; Springer: Cham, Switzerland, 2014; pp. 818–833.
32. Tang, G.; Xiao, Z.; Liu, Q.; Liu, H. A Novel Airport Detection Method via Line Segment Classification and Texture Classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2408–2412.
33. Graham, R.L.; Knuth, D.E.; Patashnik, O. *Concrete Mathematics: A Foundation for Computer Science*, 2nd ed.; Addison-Wesley Longman Publishing Co., Inc.: Boston, MA, USA, 1994.
34. Hosang, J.; Benenson, R.; Dollár, P.; Schiele, B. What makes for effective detection proposals? *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 814–830.
35. Zhao, D.; Shi, J.; Wang, J.; Jiang, Z. Saliency-constrained semantic learning for airport target recognition of aerial images. *J. Appl. Remote Sens.* **2015**, *9*, 096058.
36. Uijlings, J.R.; Van De Sande, K.E.; Gevers, T.; Smeulders, A.W. Selective search for object recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171.