

Article

A Hierarchical Airport Detection Method Using Spatial Analysis and Deep Learning

Fanxuan Zeng ^{1,2} , Liang Cheng ^{1,2,3,4,*}, Ning Li ^{1,2}, Nan Xia ^{1,2}, Lei Ma ^{1,2}, Xiao Zhou ^{1,2} and Manchun Li ^{1,2}

¹ Jiangsu Provincial Key Laboratory of Geographic Information Science and Technology, Nanjing University, 163 Xianlin Road, Nanjing 210023, China; DG1827001@smail.nju.edu.cn (F.Z.);

MG1827061@smail.nju.edu.cn (N.L.); DG1627022@smail.nju.edu.cn (N.X.); maleinju@nju.edu.cn (L.M.); DG1727039@smail.nju.edu.cn (X.Z.); limanchun@nju.edu.cn (M.L.)

² School of Geography and Ocean Science, Nanjing University, 163 Xianlin Road, Nanjing 210023, China

³ Collaborative Innovation Center for the South Sea Studies, Nanjing University, 163 Xianlin Road, Nanjing 210023, China

⁴ Collaborative Innovation Center of Novel Software Technology and Industrialization, 163 Xianlin Road, Nanjing University, Nanjing 210023, China

* Correspondence: lcheng@nju.edu.cn

Received: 19 July 2019; Accepted: 12 September 2019; Published: 20 September 2019



Abstract: Airports have a profound impact on our lives, and uncovering their distribution around the world has great significance for research and development. However, existing airport databases are incomplete and have a high cost of updating. Thus, a fast and automatic worldwide airport detection method can be of significance for global airport detection at regular intervals. However, previous airport detection studies are usually based on single remote sensing (RS) imagery, which seems an overwhelming burden for worldwide airport detection with traversal searching. Thus, we propose a hierarchical airport detection method consisting of broad-scale extraction of worldwide candidate airport regions based on spatial analysis of released RS products, including impervious surfaces from FROM-GLC10 (fine resolution observation and monitoring of global land cover 10) product, building distribution from OSMs (open street maps) and digital surface model from AW3D30 (ALOS World 3D—30 m). Moreover, narrow-scale aircraft detection was initially conducted by the Faster R-CNN (regional-convolutional neural networks) deep learning method. To avoid overestimation of background regions by Faster R-CNN, a second CNN classifier is used to refine the class labeling with negative samples. Specifically, our research focuses on target airports with at least 2 km length in three experimental regions. Results show that spatial analysis reduced the possible regions to 0.56% of the total area of 75,691 km². The initial aircraft detection by Faster R-CNN had a mean user's accuracy of 88.90% and ensured that all the aircrafts could be detected. Then, by introducing the CNN reclassifier, the user's accuracy of aircraft detection was significantly increased to 94.21%. Finally, through an experienced threshold of aircraft number, 19 of the total 20 airports were detected correctly. Our results reveal the overall workflow is reliable for automatic and rapid airport detection around the world with the help of released RS products. This research promotes the application and progression of deep learning.

Keywords: remote sensing (RS); deep learning (DL); spatial analysis; object-based target detection; airports

1. Introduction

Airports have attracted much attention in recent decades as being key transportation targets [1,2]. Uncovering global airport distribution has great significance for transport planning and analyzing

human mobility patterns, but is difficult [3,4]. With the increasing number of global airports every year, large-scale, rapid, and automated airport detection is important. The development of remote sensing (RS) and the growing availability of RS imagery with high spatial resolution have elevated global airport detection to a new height, which can precisely identify and locate airports [5,6]. However, due to the huge amount of high-resolution imagery and computational complexity of detection algorithms, it is best to first extract the data at a broad scale [7–9]. Differing from previous researches on airport detection in single RS imagery [7,9–12], a hierarchical framework has been introduced for global airport detection: broad-scale extraction of candidate airport regions based on spatial analysis of released digital products, while at a narrower scale, suitable feature descriptors should be selected to identify airports from within the candidate regions. Thus, how to efficiently obtain airports candidate regions and robustly describe airport appearances are important for improving detection performance [10].

For candidate airport regions, simpler and more efficient segmentation-based approaches are necessary for broad-scale searching, where spectral features, textural features, or geometrical characteristics are frequently evaluated, such as gradient of intensity [13], airport line features [7,11,14], and key scale-invariant feature transform (SIFT) points [15]. For the gradient of intensity, common machine learning classifiers, such as the Adaboost algorithm, were adopted for the rough identification of airport runways [13]. However, the gradient of intensity at the pixel level cannot represent the substantive characteristics, and the line features are more specific for airport regions. Budak et al. [7] proposed an algorithm composed of several line-based processing steps, Zhu et al. [14] introduced the concept of near parallelity, and Tang et al. [11] applied a line segment detector to extract the features of line segments in images. The proposed line-based methods have good performance, but also need pixel-level operation, which requires enormous computation time and memory, thus limiting their broad-scale application. Although the SIFT key-points method has a high computation efficiency with high accuracy, it is still not suitable for relatively low-resolution imagery at a broad scale [15]. In general, these studies commonly used features at the same scale, where single broad-scale features could cause the missing of detailed information and low accuracy, and single narrow-scale features could cause low computation efficiency.

The fusion of multiple features, especially at different scales, can be used to improve the extraction performance. Zhao et al. [16] fused the bottom-up region features and top-down line features to reduce the sensibility of resolution variety, and Xiao et al. [12] used multiscale fusion features to represent complementary information and proved that fusion of features outperformed other features for candidate airport region extraction. Except for characterization of two-dimensional profiles, spatial characterization of three-dimensional profiles should also be considered to improve efficiency, such as elevation differences derived from existing digital elevation models [17]. Runways, with their unique textural and spatial characteristics, are most prominently chosen to rapidly locate candidate airports [7,8,11]. Here, runways refer specifically to the part of airport used for take-off and landing, and can be directly extracted from impervious areas of land cover products, such as FROM-GLC10 (fine resolution observation and monitoring of global land cover 10) [18]. Moreover, they have a distinct shape, flat terrain, and are narrow and long with no direct connection to public roads. These unique features allow spatial analysis methods to distinguish airports from other land uses.

At the narrow scale, other unique features are used to distinguish airports, including parking lots, terminals, hangars, and aircraft [10]. The latter's appearance is uniform in different locations, making them easy to detect at a narrower scale [19]. Aircraft detection is relatively mature compared with other targets. The many methods of aircraft detection can be roughly classified into two categories [20]: low-level features, such as edges and symmetry [21–27], and high-level features based on object features [20,28–33]. For low-level features, Bo et al. [21] converted RGB images to binary images for aircraft detection. The image conversion, essentially, is a dimension reduction method, which loses some descriptive features on the premise of reducing computational cost. Compared with the dimension reduction method, Luo et al. [22] trained a support vector machine (SVM) classifier based on

histogram of oriented gradient (HOG) features. The low-level aircraft features detector has acceptable accuracy with low computational complexity, but has poor robustness.

For high-level features, by increasing the dimensions of spectral features, textural features, or geometrical characteristics, classification stability is significantly improved [28,32,34,35]. Deep learning (DL) is the most prominent method that can automatically learn high-level features with high accuracy, and has created new ways to analyze remote sensing imagery [36]. With the deepening of computer vision research, convolution feature extraction was proposed due to its strong representation ability. Convolutional neural networks (CNN), a kind of mature DL algorithm, have achieved significant success in target detection with the help of prior knowledge-based region proposal methods [29,30,37]. For the prior knowledge, low-level features [37] and pretrained convolutional networks [29,30] are commonly used. However, the region proposal method largely relies on supervised pretraining, which increases the computation complexity, and thus a selective search was introduced in the R-CNN (regional-CNN) and Fast R-CNN framework that required no prior knowledge [38,39]. Moreover, inefficient selective searching can be replaced by a more effective region proposal network which requires no prior knowledge and has better performance, called Faster R-CNN [40,41]. This Faster R-CNN framework has proven successful and efficient for aircraft detection [42–45].

The detection of airports around the world is important but challenging due to variations in airport appearance and size, and the complexity of their backgrounds. In this paper, we propose a hierarchical airport detection method that obtains candidate airport regions based on released remote sensing products, then uses high-resolution imagery to detect aircraft and distinguish airports from other impervious surfaces, such as residual segmented roads or buildings. In general, the main contributions of this work can be summarized as:

(1) proposal of a worldwide airport detection workflow based on mature DL methods and spatial analysis of released digital products with the advantages of being fast and automatic due to the omission of processing and analyzing original remote sensing data.

(2) the integration of DL and spatial analysis for global airport detection as an exploration of the in-depth application of DL in the field of object detection, which has been urgently called for in a previous publication [36].

2. Materials

In this study, three experimental areas were selected to test the proposed method as shown in Figure 1, including Beijing (China), New Jersey (U.S.), and northeastern Buenos Aires (Argentina), with areas of 16,394 km², 38,491 km², and 20,806 km² respectively. The selection of three experimental areas is due to consideration of geographical diversity, which can be used to verify the universality and reliability of the proposed workflow. Major datasets used included the global land cover (GLC) product FROM-GLC10, global digital surface model ALOS World 3D—30 m (AW3D30), open street maps (OSMs) for roads and buildings, shape-files for administrative boundaries, and high-resolution imagery from Google Maps (Table 1).

The FROM-GLC10 included 10 land cover types with 10 meter spatial resolution and high overall accuracy [18]. Compared with other GLC products, FROM-GLC10 has the characteristics of high spatial resolution and global coverage, which are conducive to the worldwide application of the workflow. This product's "impervious area" category was used to locate possible airport runways [46]. The AW3D30, provided by the Japan Aerospace Exploration Agency, has 30 meter horizontal resolution and higher accuracy than ASTER, SRTM1, and SRTM3 [47,48]. This was used to remove the nonground features in impervious areas. OSM datasets provided widely used building distribution and road network products, which could help eliminate nonrunway areas and segment the surface into blocks for distinguishing airport runways. [49]. In order to validate the reliability of the proposed workflow, airport validation data were downloaded from <http://ourairports.com/data/>, which included the locations and descriptions of some existing airports.

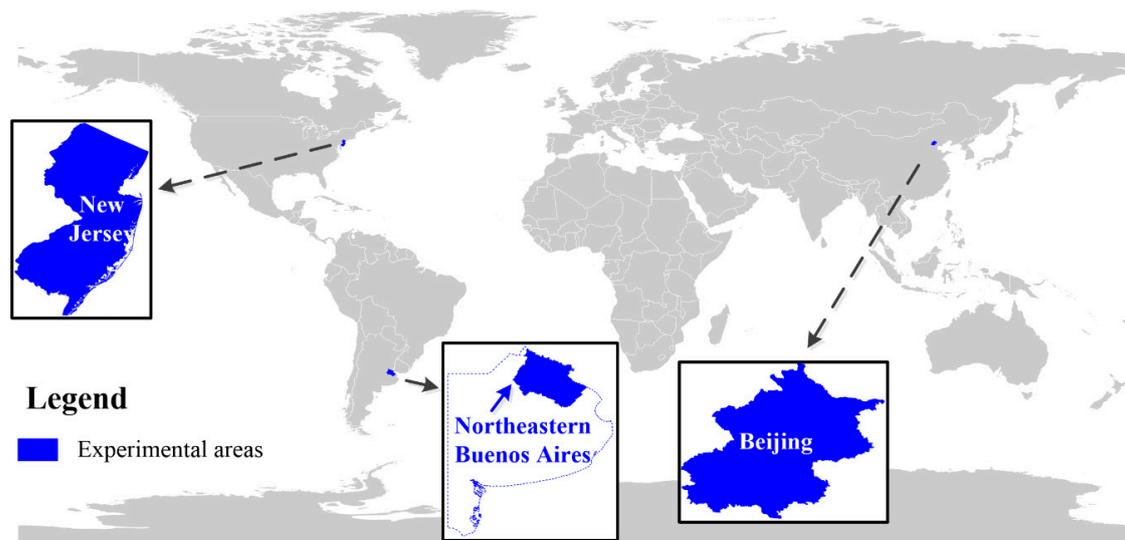


Figure 1. The experimental areas included Beijing, New Jersey, and northeastern Buenos Aires, in Asia, North America, and South America, respectively.

Table 1. Data sources.

Datasets	Spatial Resolution (m)	Updated	Sources
FROM-GLC10	10	2017	Tsinghua University
AW3D30	30	April 2019	Japan Aerospace Exploration Agency
OSM datasets	-	May 15th, 2019	Open Street Map
Administrative boundaries	-	-	Center for Spatial Sciences, University of California, Davis
Google images (19-level)	0.23	-	Google
Airport validation data	-	June 12th, 2019	Crowd-sourced

3. Methods

The proposed workflow for airport detection first defines candidate regions and then detects aircraft (Figure 2). There were three steps for identifying candidate airport regions, included exclusion of nonground regions, block segmentation by road networks, and blocks extraction with area and length thresholds. For the detection of aircraft, Faster R-CNN was used and a second CNN classifier was used for refining the aircraft detection.

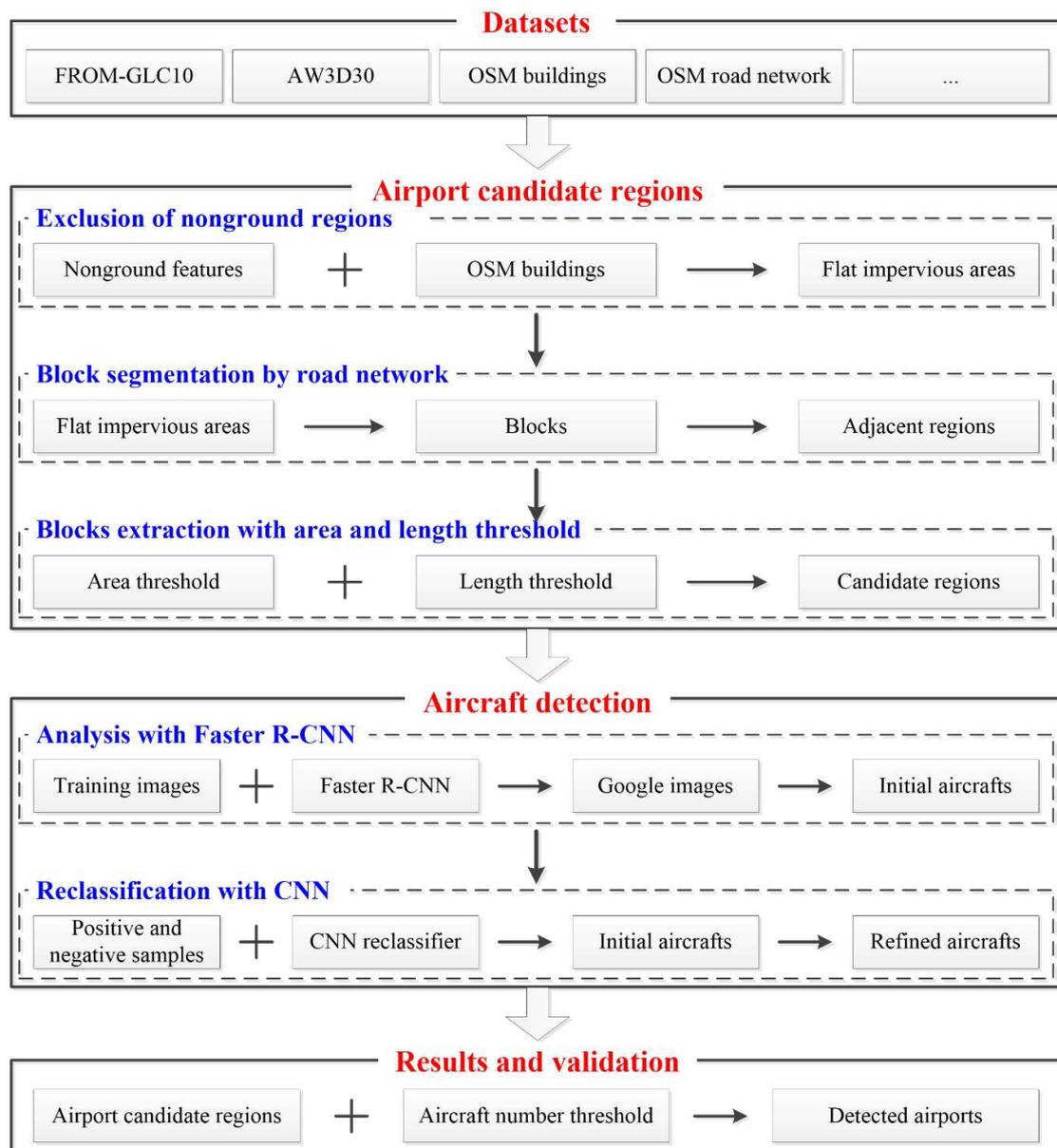


Figure 2. Workflow of the proposed airport detection method.

3.1. Candidate Airport Regions

3.1.1. Exclusion of Nonground Regions

Impervious area in the FROM-GLC10 is composed of two parts: the tops of artificial structures (nonground) and the ground itself, which need to be separated for accurate classification [46]. A morphological filter can be used to extract nonground features from the AW3D30 dataset [50]. For every step, using a sliding window (2×2 pixels) to extract the minimum elevation grid allows grids with a height difference exceeding a given threshold to be regarded as nonground features (Figure 3). We set this threshold conservatively as 10 meters. The buildings and nonground features can be excluded as possible airport runways, and erasing them reduces the candidate areas and generates flat impervious regions for the next step.

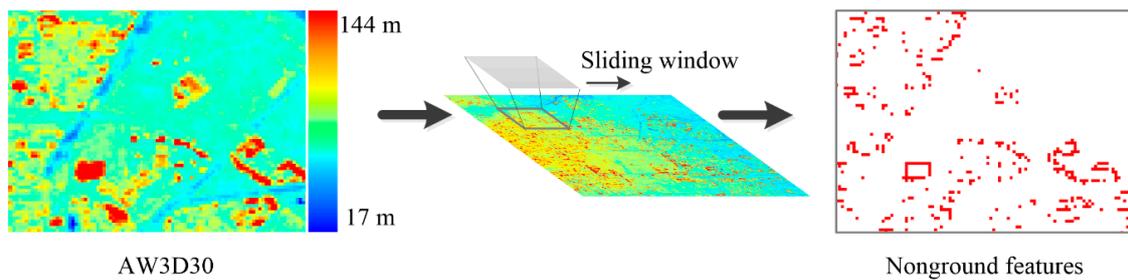


Figure 3. Morphological filtering to extract nonground features. The “AW3D30” was colored to show elevation changes.

3.1.2. Block Segmentation by Road Networks

Song et al. proposed recognizing city blocks using road networks, allowing impervious urban areas to be segmented by erasing road networks [51]. In OSM roads, airports’ internal roads are defined as “service roads”, which we retained to maintain connections between runways and parking lots. Road networks were treated as a 10-meter buffer and segmented the flat impervious regions into blocks, which was converted to shapefile format. After block segmentation, the airport is separated from other blocks and appeared spatially nonadjacent. This allows spatial clustering based on adjacency analysis to group adjacent regions (Figure 4). All the processing was finished with the help of the ArcMap 10.3 platform.

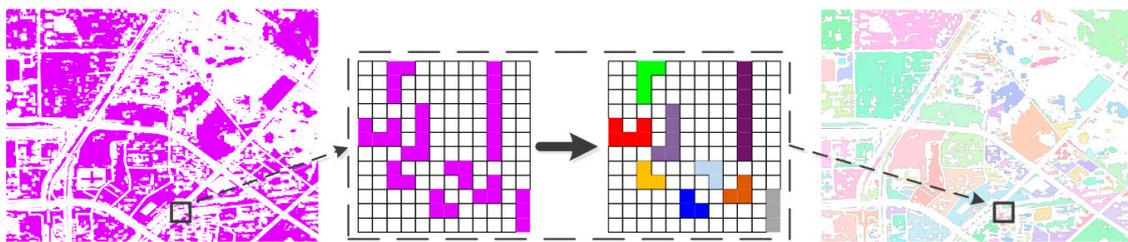


Figure 4. Spatial clustering based on adjacency analysis, in which different colors represent different groups.

3.1.3. Blocks Extraction with Area and Length Thresholds

For extracted blocks, those with small areas can be first removed by defining an area threshold. According to our surveys and statistics, the area of an airport is usually larger than 0.1 km^2 ; therefore, an area threshold can greatly decrease the calculation of the next steps. Then, the length threshold was applied to represent the features of the airport runway (Figure 5), which is defined by the diameter of a minimum circumscribed circle. The maximum length of an individual block is determined by the size of the airport, and larger airports usually have longer runways. In this paper, medium and large airports were regarded as the target, and they usually have a block longer than 2 km. Thus, a 2 km length threshold was selected. The area and length of each block were measured using the ArcMap 10.3 platform.

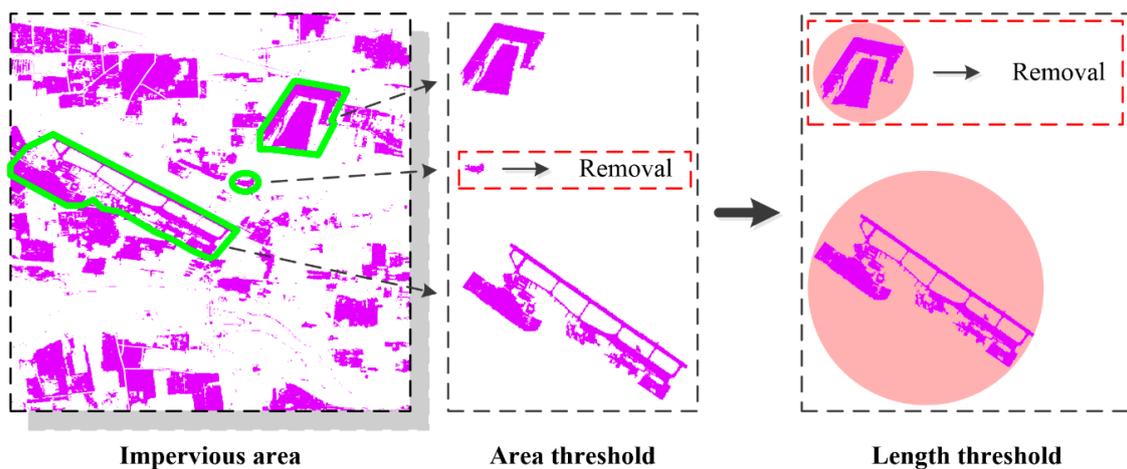


Figure 5. Selection of a candidate airport region.

3.2. Aircraft Detection

The 2D convex hull for each group is the boundary of the candidate airport region; high-resolution Google images were downloaded according to the boundary vector. Then, the Faster R-CNN was used to detect aircraft in high-resolution images.

For training purposes, we selected 50 global airports and downloaded 19-level Google images (2048×2048 pixels) with 0.23 meter spatial resolution as training images for Faster R-CNN. The selected 50 airports did not include any airports in the three experimental areas. Usually, machine learning methods require enormous training samples. Replicates were produced by rotating the images by 90, 180, and 270 degrees to further increase the number of training images. After rotation, there were 200 total training images containing 1360 labeled aircraft for the Faster R-CNN training process.

3.2.1. Analysis with Faster R-CNN

The Faster R-CNN is composed of two modules, the region proposal network (RPN) and the Fast R-CNN module. Aircraft are relatively small objects when compared with the whole candidate region. Smaller objects decrease the efficiency of RPN module; thus, the detection accuracy of Faster R-CNN drops as the object's relative size becomes smaller [40,52]. Although research on small object detection has advanced in recent years [53], processing full-size images (billions of pixels) still results in poor performance and high computational cost.

We used sliding windows to segment the full-size candidate airport images and independently detected parts with multithreading (Figure 6). Here, the sliding window's length was 2048 pixels. The stride of the sliding window should be less than the window's length to make sure there is some overlap between the two windows. The overlap ensures that aircraft located in the boundary of two neighboring windows can be detected correctly. According to the size of aircraft and the imagery resolution, a 1792-pixel window stride chosen, resulting in a 12.5% overlap area. The subimages with aircraft prediction results were the outputs of the Faster R-CNN. To detect as many aircraft as possible, a 0.5 probability threshold was selected, but this resulted in substantial overestimation in which many background regions were wrongly classified as aircraft. Thus, we applied a second classifier to reduce such overestimation [54].

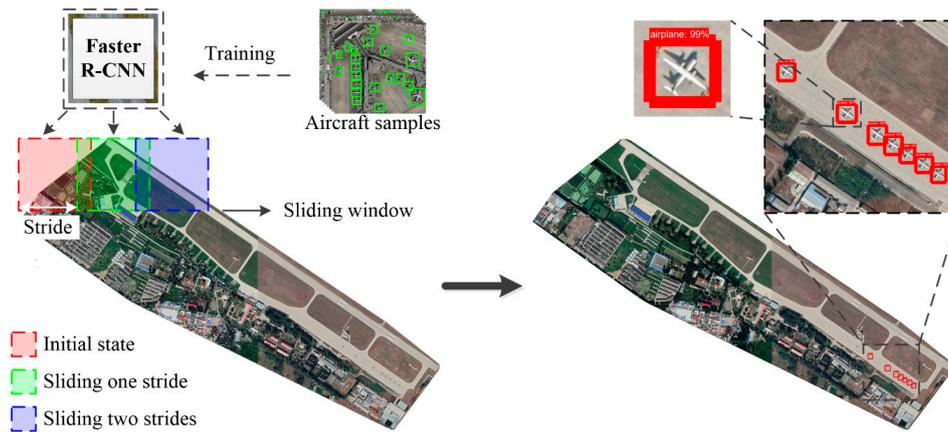


Figure 6. Strategy for aircraft detection from the full-size candidate airport images with Faster R-CNN.

3.2.2. Reclassification with CNN

Considering the overestimated outputs of Faster R-CNN, a second state CNN classification was adopted in this workflow. The GoogLeNet was selected as a CNN reclassifier for better performance [12], and all regions identified as aircraft by Faster R-CNN were used as testing samples.

For training the CNN reclassifier, there are positive training samples and negative training samples for refining outputs of Faster R-CNN (Figure 7). For positive training samples, there are 1360 subimages clipped from Faster R-CNN training images. For negative training samples, 8000 images (2000 randomly selected sites after rotation) were selected, excluding the three study regions, and with 0.23 meter spatial resolution and 2048 × 2048 pixels. They were all predicted by the previously trained Faster R-CNN, and 1072 prediction results (subimages) were regarded as nonaircraft samples, namely, the negative samples. The global positive samples used in Faster R-CNN training (1360 aircraft samples) and global negative samples detected by Faster R-CNN (1072 nonaircraft samples) composed the training samples of CNN reclassifier. The trained CNN reclassifier was applied to refine the class labeling for aircraft detection by Faster R-CNN, and reclassification results were called refined aircraft.

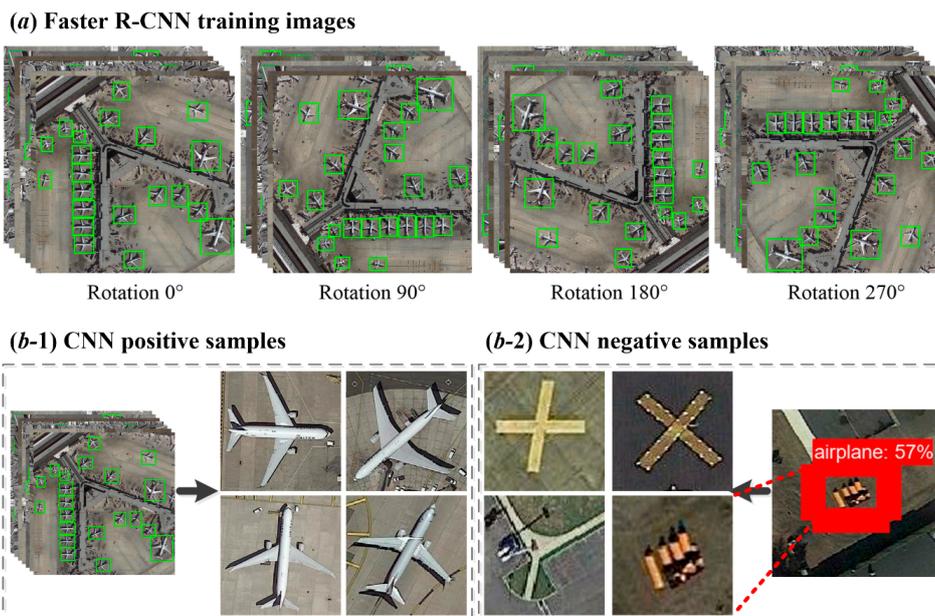


Figure 7. Examples of training samples. (a) Training images for Faster R-CNN; (b-1) positive samples and (b-2) negative samples for the CNN reclassifier.

4. Results and Validation

We considered medium and large airports with blocks at least 2 km long as targets. The candidate regions were reduced to less than 2% of their original area after spatial analysis; this process was mainly influenced by the size of the target airports and the regional urbanization level (Figure 8). The initial impervious surface was larger for Beijing than New Jersey or northeastern Buenos Aires, resulting in larger candidate regions in the former than in the latter two (Table 2).

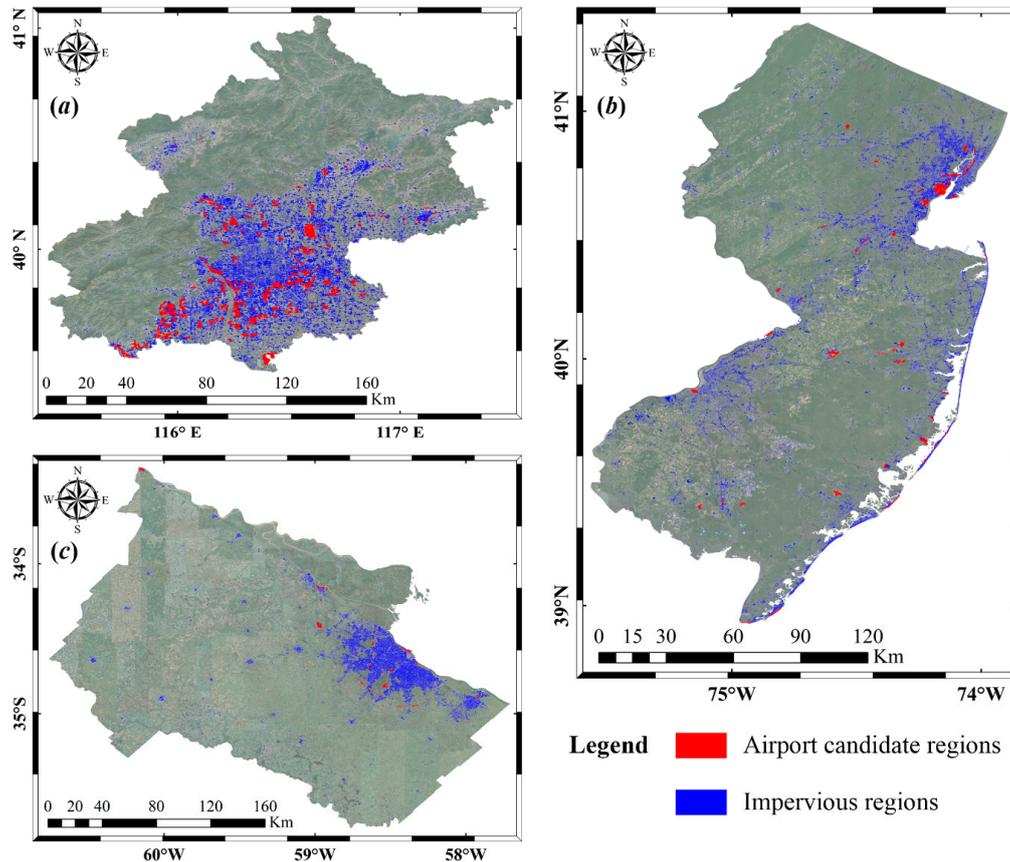


Figure 8. Candidate airport regions: (a) Beijing; (b) New Jersey; (c) northeastern Buenos Aires. The blue masks were extracted from FROM-GLC10 products, and background images were downloaded from Google Maps.

Table 2. Characteristics of target candidate airport regions where target airports had blocks 2 km long. “All impervious area” and “Candidate airport impervious area” represent blocks before and after spatial analysis, respectively. “Relative area” means the proportion of block area in each experimental area, and “block counts” represents the number of blocks before and after spatial analysis.

City	Attribution	Impervious Area	
		All	Candidate Airport
Beijing	Relative area (%)	15.49	1.64
	Block counts	294,955	192
New Jersey	Relative area (%)	7.95	0.37
	Block counts	537,640	58
Northeastern Buenos Aires	Relative area (%)	5.05	0.06
	Block counts	164,266	23

Beijing had the highest artificial surface coverage (15.49%) with fewer blocks than New Jersey (Table 2), which reflects the larger average area of blocks in Beijing than in New Jersey. Besides, there was better constraint of spatial analysis in northeastern Buenos Aires, which decreased the relative area and blocks to a low level.

For the Faster R-CNN training, the initial learning rate was 0.001 and halved with every 20,000 iterations. The batch size was 1, the training step was 60,000, and the momentum optimizer value was 0.9. The output of the Faster R-CNN is essentially a set of relative coordinates of aircraft proposals in the input image. Each proposal is predicted to contain aircraft with an estimated probability greater than a predefined threshold. With the 0.5 probability threshold, 829 proposals (737 aircrafts and 92 nonaircrafts) were predicted with 88.90% mean user's accuracy in all three experimental areas (Table 3). This output was fed into the second-state CNN automated classifier. The training samples were classified into two classes, namely, the aircraft (positive samples) and nonaircraft (negative samples). The GoogLeNet learning rate was 0.001 and was halved for every 20,000 iterations. The batch size was 16 and the training step was 100,000. The user's accuracy improved significantly after CNN reclassification, and 760 refined results included 44 nonaircrafts (Table 3). The results of aircraft detection for candidate regions are shown in the subgraphs of Figure 9; different subimages are from different candidate airport regions, and the detection distinguished airports from the background. The green line represents the boundary of candidate airport regions and red boxes are the proposed aircraft regions detected by Faster R-CNN.

Table 3. Accuracy of aircraft prediction with Faster R-CNN and the improvement after the CNN was applied to the predictions from the Faster R-CNN.

City	Faster R-CNN		Refined by CNN Reclassifier	
	Prediction	User's Accuracy	Prediction	User's Accuracy
Beijing	340	90.88%	313	95.53%
New Jersey	380	88.16%	348	93.68%
Northeastern Buenos Aires	109	85.32%	99	91.92%
Total	829	88.90%	760	94.21%

As shown in Table 3, 760 refined results with 94.21% user's accuracy included 716 aircrafts and 44 nonaircraft, and all of the 760 results are located in 29 blocks. In order to filter the false airport detections caused by the 44 nonaircraft, we filtered the 10 out of 29 testing regions that contained less than seven aircraft in the CNN results (white dots in Figure 10), where seven is the average aircraft number of the global 50 airports (1360/200). Finally, 19 airports (red dots in Figure 10) in the three experimental regions were identified.

Airport validation data were used to validate the reliability of the proposed workflow. As errors or incomplete records were common in this dataset, cross-validation was adopted. Of the 20 airports with blocks longer than 2 km, 19 were detected correctly. The missing airport was located in northeastern Buenos Aires (Figure 10c), where the impervious areas in the original FROM-GLC10 product did not contain a complete runway for this airport, resulting in missing data during the subsequent processing. In the experimental areas, no airports were found that were not in the existing database.

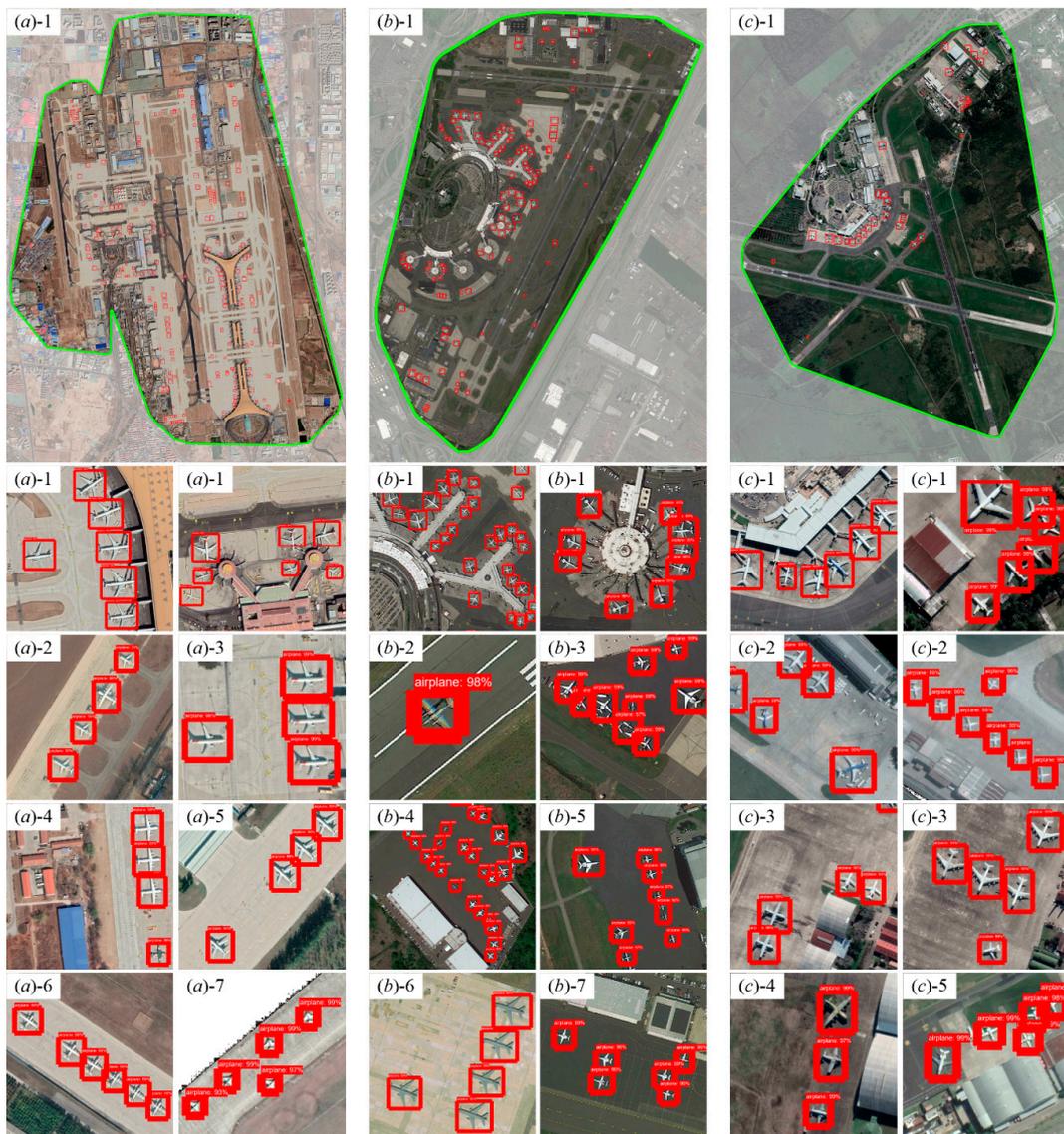


Figure 9. Some examples of aircraft detection results. Different subimages are from different candidate airport regions and the numbers record the airports’ serial number. (a) seven airports in Beijing; (b) seven airports in New Jersey; (c) five airports in northeastern Buenos Aires. The green line represents the boundary of candidate airport regions and red boxes are the detected aircraft.

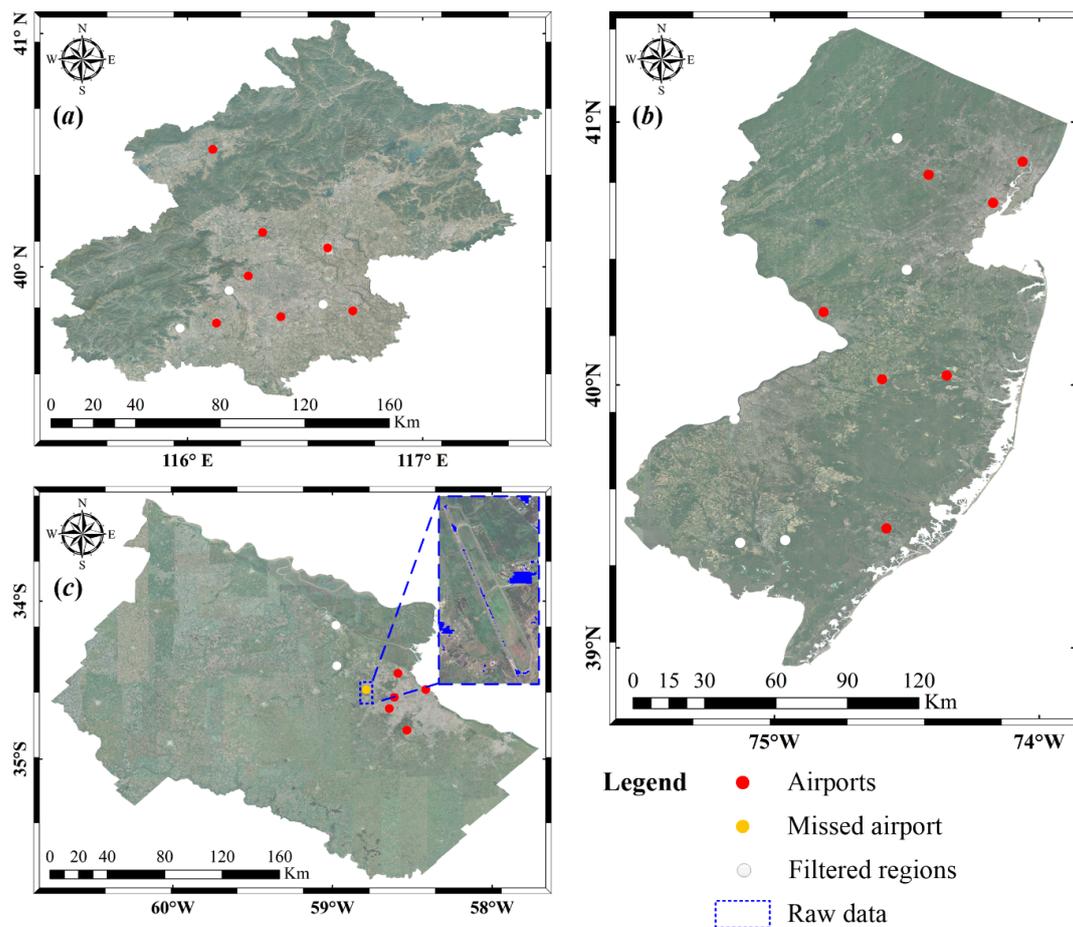


Figure 10. Airport detection results. (a) Beijing; (b) New Jersey; (c) northeastern Buenos Aires. Background images were downloaded from Google Maps.

5. Discussion

5.1. Candidate Airport Regions

5.1.1. Possibility of Worldwide Airport Detection from FROM-GLC10 Product

In Figure 10c, the incompleteness of impervious areas is caused by the accuracy of the FROM-GLC10 product, which is mainly influenced by the method and original data used in product generation. For the missing airport, the runway was covered with vegetation, which meant that the spectral characteristics were similar between the runway and the grassland. However, the 10 meter spatial resolution of FROM-GLC10 is still the optimal product for a narrow runway, with which 19 airports were detected correctly. In these 19 airports, impervious areas covered airport runways completely by visual interpretation, and the overall coverage rate reached 95% (Figure 11). Compared with the released products, the generation of impervious areas using raw remote sensing data is high-cost and decreases the efficiency of worldwide airport detection. In summary, global airport detection by the FROM-GLC10 product is viable and efficient. In the future, improved GLC products should also be discussed for the improvement of global airport detection.

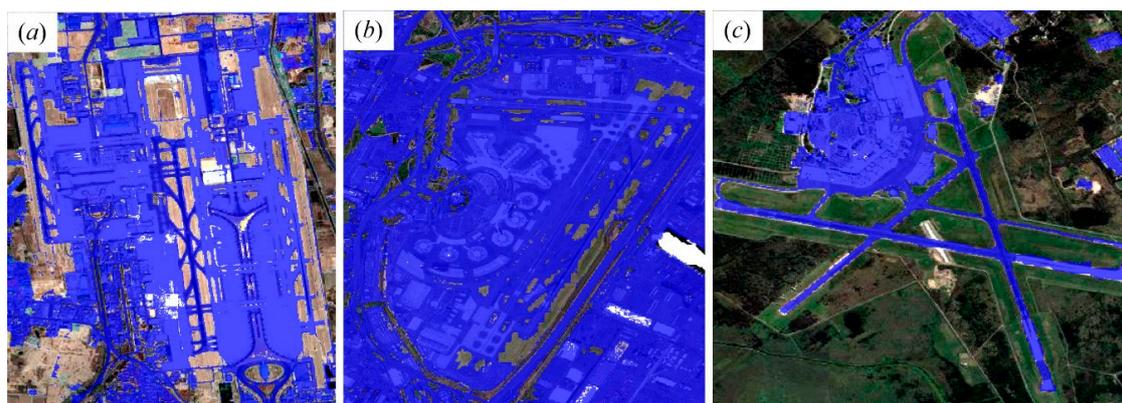


Figure 11. Samples of FROM-GLC10 product coverage of airports. (a) Beijing; (b) New Jersey; (c) northeastern Buenos Aires. Blue masks represent the impervious areas. Background images were downloaded from Google Maps.

Generating a synergetic GLC map by fusion of different GLC products is a novel thought [55]. In this paper, only FROM-GLC10 was used instead of a fusion of multisource GLC products, as different products have various spatial resolutions, date of production, and classification system, such as the MODIS product (MOD12Q1) with 1 km spatial resolution and all 17 classes of the IGBP legend [56]. The fusion of various GLC products requires different methods and data sources, which inevitably increase the time and computational expense. Even so, it is debatable whether the integration would improve the accuracy. In future research, different GLC products should be compared and the fusion of these products should be systematically discussed.

5.1.2. Selection of Area and Length Thresholds

The selected threshold in spatial analysis has a great impact on block segmentation results: lower thresholds produce more unnecessary blocks for further analysis, while higher thresholds cause more airports to be missed. We analyzed the statistics of the number of blocks and airports detected with different thresholds (Figure 12). It is clear that the length threshold had a stronger effect than the area threshold. The optimal parameters, as shown in Figure 12, retain the most airports with the fewest blocks. For northeastern Buenos Aires, the original thresholds were optimal, while for Beijing, the optimal thresholds were 0.14 km² and 2.2 km. However, as New Jersey airports were numerous and variable in size, it was more difficult to choose an appropriate threshold that constrains all airports. Country development levels and government policy together influenced the airport size and number [57]. From a global perspective, the choice of area and length threshold were suitable and generalized, but need to be adjusted appropriately for specific areas.

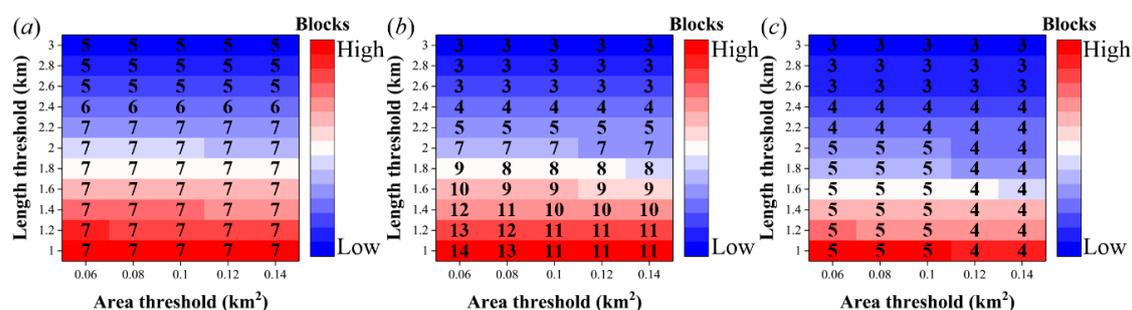


Figure 12. Airports and remaining blocks with different area and length thresholds. The warm color represents more blocks than the cool color, and the numerals record the number of airports. (a) Beijing; (b) New Jersey; (c) northeastern Buenos Aires.

5.2. Aircraft Detection

5.2.1. Parameters of the Sliding Window in Faster R-CNN

The presence of agminated aircraft was an important basis for distinguishing airports from other candidate regions. Thus, it was necessary to detect as many aircraft as possible. A sliding window searching strategy was adopted for aircraft detection from the full-size candidate airport images with Faster R-CNN. There were two parameters in the processing: the sliding window's length and stride. The length of the sliding window in this study was 2048 pixels, which was the same as the length of the training images and ensured that the relative size of aircraft was constant. For images with the same resolution, enlarging the length of the sliding window would reduce the relative size of aircraft. Previous research suggested that smaller objects decrease the efficiency of the RPN module, thus, the detection accuracy of Faster R-CNN dropped as the object's relative size became smaller [52].

The stride of the sliding window should be smaller than the window's length to make sure there is some overlap between two adjacent windows. The overlap ensured aircrafts located in the boundary of two neighboring windows could be detected correctly. We adjusted the window's stride, which varied from 1152 pixels to 2048 pixels with a 128-pixel interval. Three airports (Figure 9a-1, Figure 9b-1, and Figure 9c-1) were taken as examples distributed in three experimental areas and the total number of aircraft was 374. We calculated the recall and accuracy of aircraft detection after final CNN refining with different strides (Figure 13). It is clear that the recall decreased with increasing window stride and the accuracy is stable.

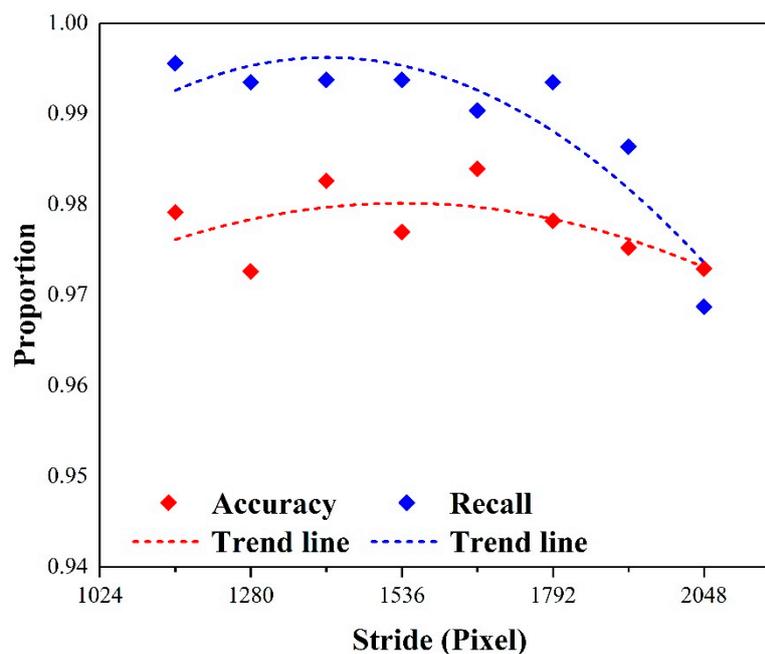


Figure 13. Influence of the sliding window's stride on aircraft detection. Accuracy represents the proportion of aircraft in all regional proposals, and recall is the proportion of detected aircraft.

5.2.2. CNN Reclassifier for Accuracy Improvement

We not only aimed at detecting as many aircraft as possible, but also at achieving high accuracy. A 0.5 probability threshold was selected in order to detect as many aircraft as possible, but this resulted in substantial overestimation in which many background regions were wrongly classified as aircraft. Therefore, two-state CNN classifiers were trained for aircraft refining. In this paper, the CNN classification was simplified as there are only two classes, namely, aircraft and nonaircraft. Results show that the CNN reclassifier is effective for refining the Faster R-CNN output. For worldwide airport

detection, as more new cities are detected using Faster R-CNN, the reclassifier training samples can be expanded, further improving the overall accuracy.

To improve the identification accuracy, previous publications have focused on refinements of the network structure used in the Faster R-CNN and CNN classifier. Most of them have been proved to be effective, such as the single-shot multibox detector (SSD) [58] and you only look once (YOLO) [59]. All of these network structures could be tested and compared to determine the best detector. Not only network structure can be explored; multiview images could also be considered, such as Google street view (GSV). Previous publications have proposed scene classification and object detection using GSV [60–62]. Thus, GSV, as an available and free resource, has potential for building an excellent airport detector in future work.

5.3. Pros and Cons

For the purpose of fast and automatic worldwide airport detection, we proposed a released RS product-based airport detection workflow. The possibility of airport detection from FROM-GLC10 product has been discussed in Section 5.1.1. Here, we discuss the pros and cons of the proposed workflow. The workflow has the following merits: (a) the spatial analysis method is effective for extracting global candidate airport regions with decreased data volume and calculating costs; (b) extracting the visual information from high-resolution RS imagery by deep learning is one of the best ways to obtain information about geographical objects. For instance, the blocks containing aircraft mainly belong to airports; (c) in the workflow, several datasets were included, such as FROM-GLC10, OSM road networks, OSM buildings, and global DSM. The worldwide candidate airport regions can be constrained with FROM-GLC10, OSM provided 2D geographical information, and DSM provides the third dimension. Thus, compared with previous studies, we integrated several released products in order to avoid processing and analyzing original remote sensing data directly, which improved the efficiency of the proposed workflow for the purpose of automatic and fast worldwide airport detection.

The proposed methods face the following problems: (a) the impacts of inaccurate products. The proposed method relies significantly on FROM-GLC10 products, OSM datasets, and global DSM, and error accumulates with the integration of these products, which has a great impact on the accuracy of airport detection; (b) The time differences among products. Due to the limitation of data acquisition, the remote sensing imagery and products were collected at different times (the FROM-GLC10 product was updated in 2017 and high-resolution remote sensing imagery was collected in 2019). Thus, newly built airports might not be detected. In a future study, a quick registration and partial update method should be explored for the FROM-GLC10 product. Moreover, research on GLC has always been a hot issue and improved GLC products are released every year. Using GLC products with higher resolution and precision can effectively improve the detection results of the proposed method in the future.

6. Conclusions

In this paper, we presented a hierarchical method for fast and automatic airport detection around the world, including broad-scale detection of impervious runway surfaces and narrow-scale detection of aircraft. Previous studies focused on the airport detection algorithm in remote sensing imagery. In this approach, impervious areas were identified through GLC products, then spatial analysis was used to better constrain the candidate airport regions. Nonground regions were extracted and removed with the help of DSM and OSM buildings. After that, the ground regions were segmented by the OSM road network. Finally, spatial cluster-based adjacency analysis was used to constrain the candidate airport regions. In the first step, the candidate airport regions were extracted by analyzing the geometrical characteristics of segmentation blocks. In the second step, the Faster R-CNN method was used for aircraft detection with 88.90% mean user's accuracy, which could be improved to 94.21% using a CNN reclassifier; areas containing many aircrafts were then defined as airports. The experimental areas contained 20 airports with blocks longer than the 2 km threshold, 19 of which were detected correctly. The missing airport can be attributed to the quality of the GLC map. Thus, the overall workflow is

reliable and can be improved further through the use of higher quality GLC product and more efficient network structure of DL.

Author Contributions: F.Z. and L.C. conceived the research idea and designed the experiments. N.L., N.X. and X.Z. assisted with the experimental result analysis. F.Z. wrote the manuscript. L.M. and M.L. edited the manuscript.

Funding: This work is supported by the National Key Research and Development Plan (2017YFB0504205), the National Science Foundation of China (41622109, 41371017, 41701374) and the Natural Science Foundation of Jiangsu Province of China (BK20170640).

Acknowledgments: Sincere thanks are given for the comments and contributions of anonymous reviewers and members of the editorial team.

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

References

1. Liu, D.; He, L.; Carin, L. Airport Detection in Large Aerial Optical Imagery. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Montreal, QC, Canada, 17–21 May 2004; pp. 761–764.
2. Wang, Y.; Pan, L. Automatic Airport Recognition Based on Saliency Detection and Semantic Information. *ISPRS Int. J. Geo-Inf.* **2016**, *5*, 115. [[CrossRef](#)]
3. de Neufville, R. Airline network development in Europe and its implications for airport planning. *Eur. J. Transp. Infrastruct.* **2008**, *8*, 264–265.
4. Bajardi, P.; Poletto, C.; Ramasco, J.J.; Tizzoni, M.; Colizza, V.; Vespignani, A. Human Mobility Networks, Travel Restrictions, and the Global Spread of 2009 H1N1 Pandemic. *PLoS ONE* **2011**, *6*, e16591. [[CrossRef](#)]
5. 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. [[CrossRef](#)]
6. Chen, Y.; Li, W.; Sakaridis, C.; Dai, D.; Van Gool, L. Domain Adaptive Faster R-CNN for Object Detection in the Wild. *2018 IEEE/Cof Conf. Comput. Vis. Pattern Recognit.* **2018**, 3339–3348.
7. Budak, U.; Halici, U.; Sengur, A.; Karabatak, M.; Xiao, Y. Efficient Airport Detection Using Line Segment Detector and Fisher Vector Representation. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 1–5. [[CrossRef](#)]
8. Liu, N.Y.; Cui, Z.Y.; Cao, Z.J.; Pi, Y.M.; Dang, S.H. Airport detection in large-scale SAR images via line segment grouping and saliency analysis. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 434–438. [[CrossRef](#)]
9. Chen, F.; Ren, R.; Van De Voorde, T.; Xu, W.; Zhou, G.; Zhou, Y. Fast Automatic Airport Detection in Remote Sensing Images Using Convolutional Neural Networks. *Remote Sens.* **2018**, *10*, 443. [[CrossRef](#)]
10. 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. [[CrossRef](#)]
11. Tang, G.; Xiao, Z.; Liu, Q. A Novel Airport Detection Method via Line Segment Classification and Texture Classification. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2408–2412. [[CrossRef](#)]
12. Xiao, Z.; Gong, Y.; Long, Y.; Li, D.; Wang, X.; Liu, H. Airport Detection Based on a Multiscale Fusion Feature for Optical Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1469–1473. [[CrossRef](#)]
13. Aytekin, O.; Zongur, U.; Halici, U. Texture-based airport runway detection. *IEEE Geosci. Remote Sens. Lett.* **2013**, *10*, 471–475. [[CrossRef](#)]
14. 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.
15. Tao, C.; Tan, Y.H.; Cai, H.J.; Tian, J.W. Airport detection from large Ikonos images using clustered sift keypoints and region information. *IEEE Geosci. Remote Sens. Lett.* **2011**, *8*, 128–132. [[CrossRef](#)]
16. Zhao, D.P.; Ma, Y.Y.; Jiang, Z.G.; Shi, Z.W. Multiresolution airport detection via hierarchical reinforcement learning saliency model. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2017**, *10*, 2855–2866. [[CrossRef](#)]
17. Polat, N.; Uysal, M. Investigating performance of Airborne LiDAR data filtering algorithms for DTM generation. *Measurement* **2015**, *63*, 61–68. [[CrossRef](#)]
18. Gong, P.; Liu, H.; Zhang, M.; Li, C.; Wang, J.; Huang, H.; Clinton, N.; Ji, L.; Li, W.; Bai, Y.; et al. Stable classification with limited sample: transferring a 30-m resolution sample set collected in 2015 to mapping 10-m resolution global land cover in 2017. *Sci. Bull.* **2019**, *64*, 370–373. [[CrossRef](#)]

19. Chen, Z.; Zhang, T.; Ouyang, C. End-to-End Airplane Detection Using Transfer Learning in Remote Sensing Images. *Remote Sens.* **2018**, *10*, 139. [[CrossRef](#)]
20. Xu, Y.; Zhu, M.; Xin, P.; Li, S.; Qi, M.; Ma, S. Rapid Airplane Detection in Remote Sensing Images Based on Multilayer Feature Fusion in Fully Convolutional Neural Networks. *Sensors* **2018**, *18*, 2335. [[CrossRef](#)]
21. Bo, S.; Jing, Y. Region-based airplane detection in remotely sensed imagery. In Proceedings of the 2010 3rd International Congress on Image and Signal Processing, Yantai, China, 16–18 October 2010; pp. 1923–1926.
22. Luo, Q.H.; Shi, Z.W. Airplane detection in remote sensing images based on object proposal. In Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016.
23. Wang, G.; Wang, X.; Fan, B.; Pan, C. Feature Extraction by Rotation-Invariant Matrix Representation for Object Detection in Aerial Image. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 851–855. [[CrossRef](#)]
24. Zhang, L.B.; Zhang, Y.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.* **2017**, *10*, 1511–1524. [[CrossRef](#)]
25. Yokoya, N.; Iwasaki, A. Object Detection Based on Sparse Representation and Hough Voting for Optical Remote Sensing Imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 1. [[CrossRef](#)]
26. Liu, G.; Sun, X.; Fu, K.; Wang, H.Q. Aircraft recognition in high-resolution satellite images using coarse-to-fine shape prior. *IEEE Geosci. Remote Sens. Lett.* **2013**, *10*, 573–577. [[CrossRef](#)]
27. Tan, Y.; Li, Q.; Li, Y.; Tian, J. Aircraft Detection in High-Resolution SAR Images Based on a Gradient Textural Saliency Map. *Sensors* **2015**, *15*, 23071–23094. [[CrossRef](#)] [[PubMed](#)]
28. Zhang, W.; Lv, W.; Zhang, Y.; Tian, J.; Ma, J. Unsupervised-learning airplane detection in remote sensing images. In *MIPPR 2015: Remote Sensing Image Processing, Geographic Information Systems, and Other Applications*; SPIE: Bellingham, WA, USA, 2015; Volume 9815, p. 981503.
29. Li, X.B.; Wang, S.J. Object detection using convolutional neural networks in a coarse-to-fine manner. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 2037–2041. [[CrossRef](#)]
30. Yang, Y.; Zhuang, Y.; Bi, F.; Shi, H.; Xie, Y. M-FCN: Effective Fully Convolutional Network-Based Airplane Detection Framework. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1–5. [[CrossRef](#)]
31. Zhu, M.; Xu, Y.; Ma, S.; Li, S.; Ma, H.; Han, Y. Effective Airplane Detection in Remote Sensing Images Based on Multilayer Feature Fusion and Improved Nonmaximal Suppression Algorithm. *Remote Sens.* **2019**, *11*, 1062. [[CrossRef](#)]
32. Yu, Y.; Guan, H.; Zai, D.; Ji, Z. Rotation-and-scale-invariant airplane detection in high-resolution satellite images based on deep-Hough-forests. *ISPRS J. Photogramm. Remote Sens.* **2016**, *112*, 50–64. [[CrossRef](#)]
33. Guo, W.; Yang, W.; Zhang, H.; Hua, G. Geospatial Object Detection in High Resolution Satellite Images Based on Multi-Scale Convolutional Neural Network. *Remote Sens.* **2018**, *10*, 131. [[CrossRef](#)]
34. Li, Y.; Zhang, H.; Shen, Q. Spectral–Spatial Classification of Hyperspectral Imagery with 3D Convolutional Neural Network. *Remote Sens.* **2017**, *9*, 67. [[CrossRef](#)]
35. Sellami, A.; Farah, M.; Farah, I.R.; Solaiman, B. Hyperspectral imagery classification based on semi-supervised 3-D deep neural network and adaptive band selection. *Expert Syst. Appl.* **2019**, *129*, 246–259. [[CrossRef](#)]
36. Ma, L.; Liu, Y.; Zhang, X.; Ye, Y.; Yin, G.; Johnson, B.A. Deep learning in remote sensing applications: A meta-analysis and review. *ISPRS J. Photogramm. Remote Sens.* **2019**, *152*, 166–177. [[CrossRef](#)]
37. Zhang, P.; Niu, X.; Dou, Y.; Xia, F. Airport Detection on Optical Satellite Images Using Deep Convolutional Neural Networks. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1183–1187. [[CrossRef](#)]
38. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conf. Comput. Vis. Pattern Recognit.* **2014**, 580–587.
39. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Washington, DC, USA, 7–13 December 2015; pp. 1440–1448.
40. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [[CrossRef](#)] [[PubMed](#)]
41. Zhang, L.; Lin, L.; Liang, X.; He, K. Is Faster R-CNN Doing Well for Pedestrian Detection? In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016.
42. Ren, Y.; Zhu, C.; Xiao, S. Deformable Faster R-CNN with Aggregating Multi-Layer Features for Partially Occluded Object Detection in Optical Remote Sensing Images. *Remote Sens.* **2018**, *10*, 1470. [[CrossRef](#)]

43. Ding, P.; Zhang, Y.; Deng, W.-J.; Jia, P.; Kuijper, A. A light and faster regional convolutional neural network for object detection in optical remote sensing images. *ISPRS J. Photogramm. Remote Sens.* **2018**, *141*, 208–218. [[CrossRef](#)]
44. Zhang, Y.H.; Fu, K.; Sun, H.; Sun, X.; Zheng, X.W.; Wang, H.Q. A multi-model ensemble method based on convolutional neural networks for aircraft detection in large remote sensing images. *Remote Sens. Lett.* **2018**, *9*, 11–20. [[CrossRef](#)]
45. Han, X.B.; Zhong, Y.F.; Feng, R.Y.; Zhang, L.P. Robust geospatial object detection based on pre-trained faster r-cnn framework for high spatial resolution imagery. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 3353–3356.
46. Mallick, J.; Rahman, A.; Singh, C.K. Modeling urban heat islands in heterogeneous land surface and its correlation with impervious surface area by using night-time ASTER satellite data in highly urbanizing city, Delhi-India. *Adv. Space Res.* **2013**, *52*, 639–655. [[CrossRef](#)]
47. Tadono, T.; Nagai, H.; Ishida, H.; Oda, F.; Naito, S.; Minakawa, K.; Iwamoto, H. GENERATION OF THE 30 M-MESH GLOBAL DIGITAL SURFACE MODEL BY ALOS PRISM. *ISPRS - Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *41*, 157–162. [[CrossRef](#)]
48. Yahaya, S.I.; El Azzab, D. Vertical accuracy assessment of global digital elevation models and validation of gravity database heights in Niger. *Int. J. Remote Sens.* **2019**, *40*, 7966–7985. [[CrossRef](#)]
49. Susaki, J. Adaptive Slope Filtering of Airborne LiDAR Data in Urban Areas for Digital Terrain Model (DTM) Generation. *Remote Sens.* **2012**, *4*, 1804–1819. [[CrossRef](#)]
50. Zhang, K.; Chen, S.-C.; Whitman, D.; Shyu, M.-L.; Yan, J.; Zhang, C. A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 872–882. [[CrossRef](#)]
51. Song, Y.; Long, Y.; Wu, P.; Wang, X. Are all cities with similar urban form or not? Redefining cities with ubiquitous points of interest and evaluating them with indicators at city and block levels in China. *Int. J. Geogr. Inf. Sci.* **2018**, *32*, 1–30. [[CrossRef](#)]
52. Chen, C.Y.; Liu, M.Y.; Tuzel, O.; Xiao, J.X. R-cnn for small object detection. *Lect. Notes Comput. Sci.* **2017**, *10115*, 214–230.
53. Eggert, C.; Brehm, S.; Winschel, A.; Zecha, D.; Lienhart, R. A closer look: Small object detection in faster R-CNN. In Proceedings of the 2017 IEEE Int. Conf. Multimed. Expo (Icme), Hong Kong, China, 10–14 July 2017; pp. 421–426.
54. Foody, G.M.; Ling, F.; Boyd, D.S.; Li, X.; Wardlaw, J. Earth Observation and Machine Learning to Meet Sustainable Development Goal 8.7: Mapping Sites Associated with Slavery from Space. *Remote Sens.* **2019**, *11*, 266. [[CrossRef](#)]
55. Pérez-Hoyos, A.; García-Haro, F.; San-Miguel-Ayanz, J. A methodology to generate a synergetic land-cover map by fusion of different land-cover products. *Int. J. Appl. Earth Obs. Geoinf.* **2012**, *19*, 72–87. [[CrossRef](#)]
56. See, L.; Fritz, S. A method to compare and improve land cover datasets: application to the GLC-2000 and MODIS land cover products. *IEEE Trans. Geosci. Remote Sens.* **2006**, *44*, 1740–1746. [[CrossRef](#)]
57. Paleari, S.; Redondi, R.; Malighetti, P. A comparative study of airport connectivity in China, Europe and US: Which network provides the best service to passengers? *Transp. Res. Part E Logist. Transp. Rev.* **2010**, *46*, 198–210. [[CrossRef](#)]
58. 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 2016 European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
59. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conf. Comput. Vis. Pattern Recognit. (Cvpr)* **2016**, 779–788.
60. Kang, J.; Korner, M.; Wang, Y.; Taubenböck, H.; Zhu, X.X. Building instance classification using street view images. *ISPRS J. Photogramm. Remote Sens.* **2018**, *145*, 44–59. [[CrossRef](#)]
61. Branson, S.; Wegner, J.D.; Hall, D.; Lang, N.; Schindler, K.; Perona, P. From Google Maps to a fine-grained catalog of street trees. *ISPRS J. Photogramm. Remote Sens.* **2018**, *135*, 13–30. [[CrossRef](#)]

62. Cao, R.; Zhu, J.; Tu, W.; Li, Q.; Cao, J.; Liu, B.; Zhang, Q.; Qiu, G. Integrating Aerial and Street View Images for Urban Land Use Classification. *Remote Sens.* **2018**, *10*, 1553. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).