

Article

A New Framework for Automatic Airports Extraction from SAR Images Using Multi-Level Dual Attention Mechanism

Lifu Chen ^{1,2}, Siyu Tan ^{1,2}, Zhouhao Pan ³, Jin Xing ⁴ , Zhihui Yuan ^{1,2,*} , Xuemin Xing ^{2,5} and Peng Zhang ^{1,2}

¹ School of Electrical and Information Engineering, Changsha University of Science & Technology, Changsha 410114, China; lifu_chen@csust.edu.cn (L.C.); tansiyu@stu.csust.edu.cn (S.T.); 17105020588@stu.csust.edu.cn (P.Z.)

² Laboratory of Radar Remote Sensing Applications, Changsha University of Science & Technology, Changsha 410014, China; xuemin.xing@csust.edu.cn

³ China Academy of Electronics and Information Technology, Beijing 100041, China; panzhouhao09@mails.ucas.ac.cn

⁴ School of Engineering, Newcastle University, Newcastle upon Tyne NE1 7RU, UK; Jin.Xing@newcastle.ac.uk

⁵ School of Traffic & Transportation Engineering, Changsha University of Science & Technology, Changsha 410114, China

* Correspondence: yuanzhihui@csust.edu.cn; Tel.: +86-188-7404-7534

Received: 31 December 2019; Accepted: 5 February 2020; Published: 7 February 2020



Abstract: The detection of airports from Synthetic Aperture Radar (SAR) images is of great significance in various research fields. However, it is challenging to distinguish the airport from surrounding objects in SAR images. In this paper, a new framework, multi-level and densely dual attention (MDDA) network is proposed to extract airport runway areas (runways, taxiways, and parking lots) in SAR images to achieve automatic airport detection. The framework consists of three parts: down-sampling of original SAR images, MDDA network for feature extraction and classification, and up-sampling of airports extraction results. First, down-sampling is employed to obtain a medium-resolution SAR image from the high-resolution SAR images to ensure the samples (500×500) can contain adequate information about airports. The dataset is then input to the MDDA network, which contains an encoder and a decoder. The encoder uses ResNet_101 to extract four-level features with different resolutions, and the decoder performs fusion and further feature extraction on these features. The decoder integrates the chained residual pooling network (CRP_Net) and the dual attention fusion and extraction (DAFE) module. The CRP_Net module mainly uses chained residual pooling and multi-feature fusion to extract advanced semantic features. In the DAFE module, position attention module (PAM) and channel attention mechanism (CAM) are combined with weighted filtering. The entire decoding network is constructed in a densely connected manner to enhance the gradient transmission among features and take full advantage of them. Finally, the airport results extracted by the decoding network were up-sampled by bilinear interpolation to accomplish airport extraction from high-resolution SAR images. To verify the proposed framework, experiments were performed using Gaofen-3 SAR images with 1 m resolution, and three different airports were selected for accuracy evaluation. The results showed that the mean pixels accuracy (MPA) and mean intersection over union (MIoU) of the MDDA network was 0.98 and 0.97, respectively, which is much higher than RefineNet and DeepLabV3. Therefore, MDDA can achieve automatic airport extraction from high-resolution SAR images with satisfying accuracy.

Keywords: deep learning; airport extraction; deep neural network; dense connection; attention mechanism; synthetic aperture radar

1. Introduction

Synthetic Aperture Radar (SAR) can acquire images all day and all night without being affected by the weather and light conditions [1], which is a tremendous advantage that optical remote sensing images cannot offer. Therefore, it plays an increasingly important role in military and civilian applications. The airports are strategic hubs of the national economy and key targets in military missions. It is of great practical significance to implement automatic airport detection from SAR images. Additionally, this work can facilitate the takeoff and landing of aircraft, assist air traffic management, and provide various navigation services. This work is also very helpful to reduce the false alarms generated by aircraft detection by excluding specious targets from SAR images.

Airports share considerable common features in SAR images [2]: (1) The visualization of long and straight runways, taxiways, and parking lots are mostly black in SAR images; (2) The ground, made of cement and asphalt, looks lighter than the runway in SAR images; (3) Aircraft and buildings such as terminals and hangars are shown as highlighted areas in SAR images because of their strong scattering characteristics. However, they are difficult to distinguish from the buildings around the airport, which also look highlighted. The complex airport runway area plays a pivotal role in airport detection [3]. Based on its distinct visual features in SAR images, the airport runway area was extracted to achieve automatic airport detection.

The main contribution of this paper is listed as:

- (1) A new framework for airport extraction is proposed. It includes three parts: down-sampling of the original SAR images, deep learning network for the airport extraction, and bilinear interpolation to acquire the extraction result of high-resolution SAR images. For SAR images with high-precision, down-sampling is performed to produce medium resolution (5 m–10 m resolution) SAR images, and then datasets are generated. After extracting airports of medium SAR images by the deep learning network, up-sampling is carried out to produce the results with the same size as the original SAR images with high-resolution.
- (2) A new deep neural network is presented to accomplish airport extraction from SAR images, which is the multi-level and densely dual attention (MDDA) network. It mainly contains two parts, the encoder and the decoder. The encoder employs the ResNet-101 to extract features with different levels. In the decoder, the features of different levels are fully utilized through dense connection, and then the essential features of the airport are extracted by using the CRP_Net_x (1, 2, 3) modules and dual attention fusion and extraction (DAFE) module to realize the airport extraction. In the DAFE module, the dual attention is introduced to fuse global semantic information via weighting spatial position and channels to extract more distinguishing features.
- (3) The proposed framework MDDA is implemented and the performance of airport extraction is evaluated by using large-scale Gaofen-3 SAR images with a 1-m resolution.

The remainder of the paper is as follows. Section 2 is the state-of-the-art, in which the development of airport detection and deep learning in semantic segmentation are described. Section 3 is the methodology, which elaborates on the proposed framework (MDDA) and the operating principle for airport extraction. In Section 4, the experiment is performed on the MDDA network using Gaofen-3 SAR images with a 1-m resolution including four airports, and the performance is assessed. Section 5 introduces the proposed network simply, and puts forward the future research. Finally, our conclusions are given in Section 6.

2. State-Of-The-Art

Since airports are important transportation hubs and military facilities, their detection has significant application values. Optical remote sensing images are usually utilized to detect airports [4]. However, it is impossible to obtain optical remote sensing images in bad weather (such as cloud, fog, rain), which has become an important problem restricting its wide application. In this case, the use of SAR images for airport detection has become a favorable choice.

The methods of airport detection can be roughly divided into two categories: one based on low-level features such as airport edges and geometric features, and the other based on high-level features of airport targets. For the first type of features, most researchers use the method of linear feature detection. Kou et al. [5] proposed an airport detection method from remotely sensed images based on line segment detectors, and Xiong et al. [6] presented a detection algorithm of airports from SAR images based on random transform and hypothesis testing. These methods rely on line segment detector (LSD) transform, random transform, or other transformation methods to obtain the linear edge segments of airports, which are then stitched for airport identification. These methods are simple and fast. However, linear segmentation in large-scale SAR images is time-consuming and prone to false detection. For the second type of features, airports are usually detected based on the object difference between the airport and the surrounding area. Zhu et al. [7] combined the saliency analysis model with edge detection to detect airports based on remote sensing images, and Liu et al. [8] integrated line segmentation and saliency analysis to detect airports based on SAR images. However, the airports in these experiments were all small airports with fewer types of objects and obvious edges, and the saliency model often generates more false alarm targets when applied to SAR images. To accomplish airport detection from high-resolution SAR images with large scales, Zhang et al. [9] pre-processed the original image to generate the region of interest (RoI) using adaptive threshold segmentation, and extracted the airport via the binary decision tree. This method could perform airport extraction, but was always confused by road networks.

In recent years, deep learning [10] has been widely used in various fields, especially in object detection and segmentation of optical images. It provides a good technical approach for the traditional target detection from SAR images. Therefore, the target detection network or image segmentation network based on deep learning can be incurred to implement airport detection. Among them, some popular networks have been widely applied in semantic segmentation. The so-called semantic segmentation is to label each pixel in the image with its corresponding category. Airport runway extraction from SAR images is to classify the SAR image pixel by pixel, and assign different category labels to the pixels of the airport runway area and the background area. Remarkable progress has been made in semantic segmentation using convolutional neural networks (CNNs) [11]. The fully connected layer of traditional CNNs classifies feature vectors with fixed length, so it can only accept input images of a specific size. To solve this problem, Jonathan et al. [12] proposed fully convolutional networks (FCN) for image segmentation, and Yang et al. [13] used FCN combined with conditional random field (CRF) to classify SAR images. Since FCN may produce rough segmentation results, Badrinarayanan et al. [14] proposed the SegNet network, but it has a poor segmentation performance on the edges of objects. DeepLab v1 [15] combined deep convolutional nets (DCNs) and fully connected CRF, and added hole convolution to improve the boundary segmentation effect; DeepLab v2 [16] introduced the atrous spatial pyramid pooling (ASPP) structure based on DeepLab v1 to improve the shortage of DeepLab v1 in fusing the information of different layers. RefineNet [17] was a new encoder–decoder architecture for semantic segmentation, which utilizes the ResNet_101 module in the encoder and RefineNet block in the decoder. Peng et al. [18] and Zhang et al. [19] facilitated the encoding network to extract the middle and high-level features of the image, and utilized the decoding network to merge and re-extract the features generated by the encoding network to finally implement the segmentation. There are several latest and excellent networks for further optimizing the accuracy or improving the efficiency of the segmentation results such as PSPNet [20], DeepLab v3 [21], DeepLab v3 + [22], and Auto-DeepLab [23].

Researchers have also applied deep learning to airport detection. Yu et al. [24] combined CNN based on the You Only Look Once (YOLO) model with salient features to extract airports and achieved good results. Xiao et al. [25] constructed a Google-LF network to fuse multiscale features, and then the generated features were input into support vector machine (SVM) to produce the detected airports. It accomplished airport detection from remote sensing images with complex background information, but the model was often overfit due to insufficient samples. Fan et al. [26] proposed a layered airport

detection algorithm based on spatial analysis and Faster R-CNN to achieve large-scale airport detection from optical remote sensing images. Li et al. [27] built an end-to-end airport detection model from remote sensing images based on a deep transferable convolutional neural network, which overcame the shortcomings of traditional CNN models for airport detection under complex backgrounds.

Most of the above studies have focused on optical remote sensing images, but there are very few studies on the application of deep learning methods to airport detection from SAR images as SAR images are hard to understand, and the speckle noise makes it more difficult for airport detection. However, in light of the tremendous advantages of SAR images, it is necessary to study them further for airport detection from SAR images based on deep learning.

3. Methodology

We proposed the multi-level and densely dual attention (MDDA) framework, which includes three components: down-sampling, deep learning network for features extraction, and up-sampling using bilinear interpolation. First, high-resolution SAR images are down-sampled to generate medium-resolution images, so that samples can contain adequate information about the airport. Second, the samples are input into the deep learning network, which includes the encoder and decoder. The encoder utilizes ResNet to produce four-level features, which are input into the decoder. In the decoder, the dense connection and dual attention mechanism are incurred to improve the ability of features extraction. Then, the airport extraction is performed. Finally, the results are up-sampled by bilinear interpolation to accomplish the airport detection of high-resolution SAR images.

3.1. Residual Network

The residual network (ResNet) was proposed by He et al. [28,29] in 2016, which solves the problem that the accuracy of the training set decreases with the deepening of the network, and makes the CNN no longer hindered by the number of layers. The deeper the layers, the better the expression will be. The ResNet is formed by stacking the numerous residual units, as shown in Figure 1. For a residual unit, the output is

$$y_l = x_l + F(x_l + w_l), \quad w_l = \{w_{l,k} | 1 \leq k \leq K\} \quad (1)$$

where F is the residual function, and w_l denotes the weight. x_l and y_l represent the input and output of the l -th residual unit. The activation between the two residual units is realized by the residual function. First, the residual function is used to calculate the residual of the input x_l , and then the residual is added with x_l to generate the output.

Let $x_{l+1} = y_l$, we can obtain the output of the L -th residual unit by recursively using Equation (1)

$$x_L = x_l + \sum_{i=1}^{L-1} F(x_i, w_i) \quad (2)$$

Equation (2) indicates that the output of the L -th residual unit can be expressed as the sum of the input of a shallow residual unit and the mapping of all complex residual functions in the middle, which shows the good back propagation ability of the network. Assuming the loss function of the network is α , then the back propagation can be obtained via

$$\frac{\partial \alpha}{\partial x_l} = \frac{\partial \alpha}{\partial x_L} \frac{\partial x_L}{\partial x_l} = \frac{\partial \alpha}{\partial x_L} \left(1 + \frac{\partial}{\partial x_l} \sum_{i=1}^{L-1} F(x_i, w_i) \right) \quad (3)$$

It can be seen that $\frac{\partial \alpha}{\partial x_L}$ and $\frac{\partial}{\partial x_l} \sum_{i=1}^{L-1} F(x_i, w_i)$ determine the value of the weight ω . Unless they are the opposite number of each other, the gradient cannot vanish. In fact, this case never happens in practice, so the gradient flow of the network from high to low layers is very smooth, which makes the training of deep networks possible.

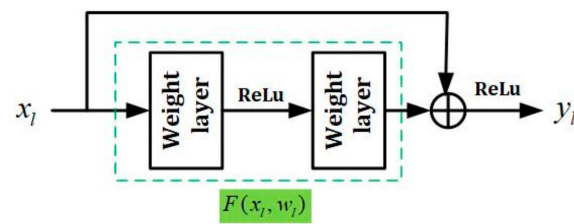


Figure 1. The structure of the residual unit.

3.2. Dense Connection

The dense connection links each layer to the others via the feedforward cascade method, as shown in Figure 2. DenseNet [30] changes the network architecture by adding the idea of skip connection and shorter connection to the residual network, and solves the problem of loss appearing or the disappearance of network input or gradient information after being transmitted through many layers. Zhang et al. [31] proposed an encoder–decoder network with dense connection to implement the extraction of water and shadow, and good results were achieved.

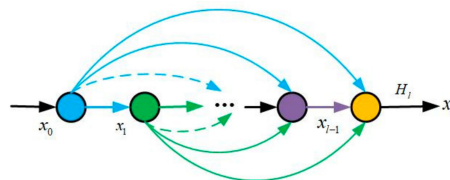


Figure 2. The dense connection.

The traditional L -layer neural network has L connections while the L -th layer network of DenseNet consists of the feature maps of the previous $L-1$ layers. Take $x_0 \dots x_{l-1}$ as input, then we can obtain

$$x_l = H_l([x_0, x_1, \dots, x_{l-1}]) \quad (4)$$

where x_0, x_1, \dots, x_{l-1} denotes the feature map connection generated by the feature maps at $0, \dots, l-1$ layers, and these connections are combined via using H_l finally.

This allows the information to be transmitted from one layer to the next, and each time, it reads information from its previous layer and writes it to the latter layer. It promotes the information transmitting of the network, strengthens the propagation of features, and enables features to be used sufficiently.

3.3. Dual-Attention Mechanism

The attention module plays an important role in the field of semantic segmentation. It weights the input feature maps, filters useful feature information, and removes redundant feature information. The attention module can fuse the input global information, and is widely used in the field of image vision. Chen et al. [32] proposed a feature recalibration network with multi-level spatial features (FRN-MSF) to implement scene classification for 11 types of scenes from SAR images, which incurred the SENet and achieved a satisfactory classification result. Fu et al. [33] extended two types of attention modules based on the self-attention module, and constructed the position attention module (PAM) and channel attention module (CAM), which work in parallel to capture the global information of the image in the spatial and channel dimensions to obtain rich contextual information.

- Position Attention Module (PAM)

The key in semantic segmentation is feature recognition. The PAM builds a positional relationship model between features by capturing global feature information, and selectively aggregates features at each position via the sum of weights for features at all positions. Regardless of the distance, similar

features will be related to each other, thus enhancing the ability of PAM to express the features. Figure 3 shows the working mechanism of the PAM.

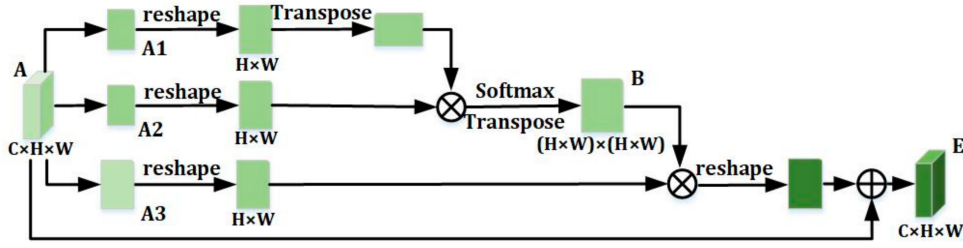


Figure 3. The position attention module (PAM).

As shown in Figure 3, the input feature map A ($C \times H \times W$) performs a convolution operation with a BN layer and a ReLU layer to produce three new feature maps A_1 , A_2 , and A_3 . They are all single-channel feature maps and they all come from feature map A , but feature maps A_1 and A_2 have the same dimensions, except for feature map A_3 . After performing the reshape operation on the feature maps of A_1 , A_2 , and A_3 , the scale of the feature maps becomes $H \times W$, then a matrix multiplication is performed on the feature of transposed A_1 and feature A_2 . B is a position attention map, and its essence is applying the softmax layer to the transposed feature map generated by the matrix multiplication.

$$B_{ij} = \frac{\exp(A_{2i} \cdot A_{3j})}{\sum_{i=1}^N \exp(A_{2i} \cdot A_{3j})} \quad (5)$$

where $N = H \times W$, and B_{ij} denotes how much the j -th position is affected by the i -th position. The more similar the position information of the two features, the larger the value of B_{ij} .

A matrix multiplication operation is performed between B and A_3 after reshaping, then D is obtained. The final output feature map E ($C \times H \times W$) is obtained by adding the reshaped D and the original feature A . Here, we need to set the weighting factor α , which is initialized to 0 and then gradually learns automatically.

$$E_j = \alpha \sum_{i=1}^N (B_{ji} \cdot A_{3i}) + A_j \quad (6)$$

It can be seen that each position of the final output feature E is a weighted sum of the features of all positions and the features of the original input, so global semantic information is aggregated.

- Channel Attention Module (CAM)

The CAM is mainly oriented to high-level features. Each channel mapping of high-level feature can be regarded as a type of response, and there is a close relationship between these types of responses. To enhance the feature map's ability to express specific semantics, the CAM obtains the interdependence relation between different channel mappings, and its working mechanism is shown in Figure 4.

Unlike the PAM, three features of A_1 , A_2 , and A_3 with the same dimensions are directly produced by reshaping the original feature A . Moreover, a matrix multiplication operation is performed on A_2 and A_1 , and then the obtained value is processed by softmax to generate the feature map X ($C \times C$).

$$X_{ij} = \frac{\exp(A_{1i} \cdot A_{2j})}{\sum_{i=1}^C \exp(A_{1i} \cdot A_{2j})} \quad (7)$$

where X_{ij} represents the effect of the i -th channel on the j -th channel.

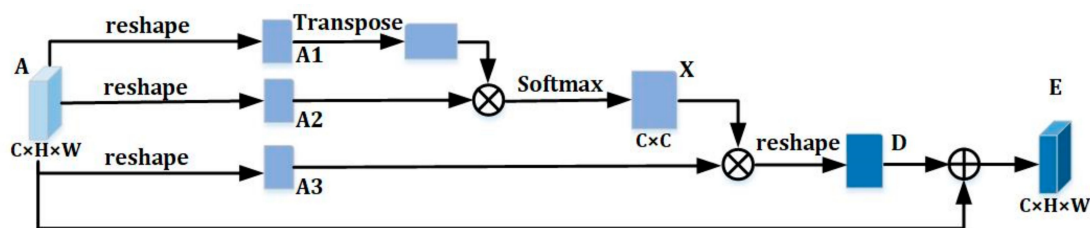


Figure 4. Channel attention module (CAM).

After a matrix multiplication of X and A3, a reshape operation is performed to generate D. Finally, a weight β is multiplied to D. Finally, the final feature map E ($C \times H \times W$) is obtained by adding the original feature A and D multiplied by the weighting factor β , which is initialized to 0 and then gradually learns automatically.

$$E_j = \beta \sum_{i=1}^C (X_{ji} \cdot A3_i) + A_j \tag{8}$$

It can be seen that the output features of each channel are the weighted sum of the features of all channels and the original features. It encodes the global semantic relationship among the feature maps of different channel and improves the ability to discriminate the feature maps.

3.4. The Proposed Automatic Airport Extraction Algorithm

To extract the airport, the multi-level and densely dual attention (MDDA) framework as shown in Figure 5 is proposed in this paper. The framework mainly includes two parts: the encoding network and decoding network. The encoding network employs the ResNet_101 [28] residual network to perform multi-level features extraction on the input dataset. The decoding network incurs a dense connection and dual attention mechanism to fuse the multi-level features and further extract essential features. It mainly consists of four modules: dual attention fusion and extraction (DAFE), CRP_Net_1, CRP_Net_2, and CRP_Net_3. The last three modules have the same internal structure. Each low-resolution feature produced by ResNet is sent to all the previous modules with higher resolution to achieve adequate fusion of features with different resolutions. After the airport segmentation is realized by extracting features from the decoding network, the up-sampling processing is carried out by bilinear interpolation to get the large-scale airport segmentation results, where the up-sampling multiple is the same as that of the input SAR image at the beginning. Finally, the airport extraction result is fused with the SAR image to generate a fusion image.

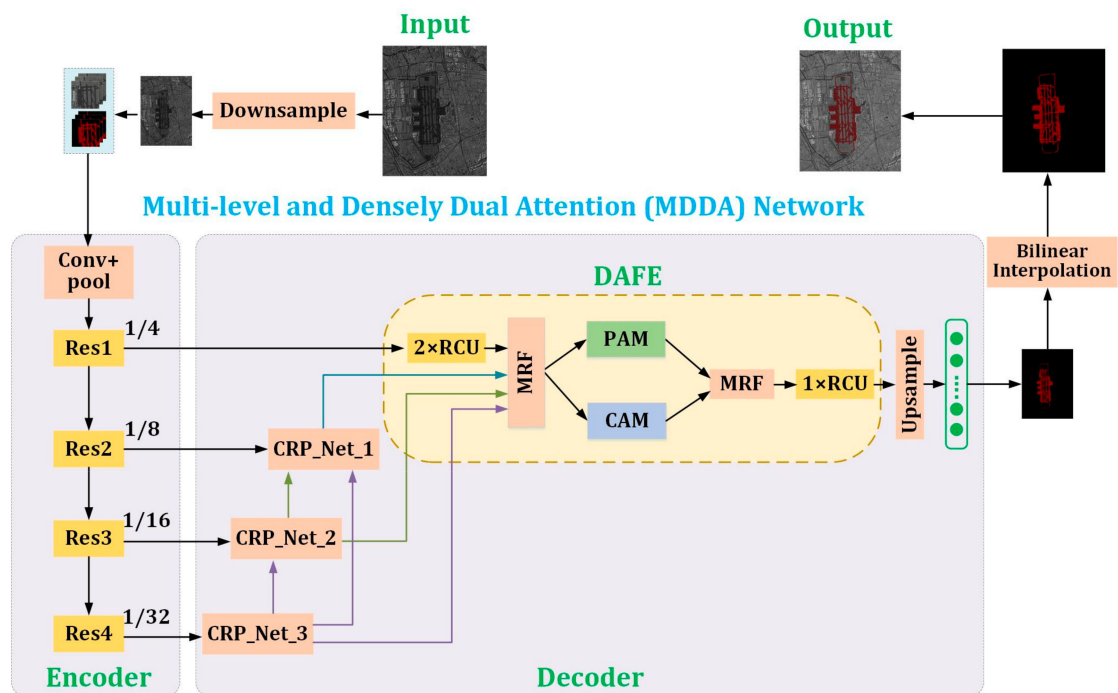


Figure 5. The multi-level densely double attention network for airport extraction.

3.4.1. Dense Connection

As shown in Figure 6a, CRP_Net_x ($x = 1, 2, 3$) is composed of a residual convolutional unit (RCU), multi-resolution fusion (MRF), and chained residual pooling (CRP). The RCU [17] is a residual unit [28] with the BN layer removed and the MRF module (as shown in Figure 6b). It consists of a series of parallel convolutions and down-samplings, which are used to fuse the features from different resolutions. The CRP (as shown in Figure 6c) [17] is the core module of CRP_Net_x . It consists of the ReLU activation function, pooling unit, and convolution unit. The features extracted from the ResNet network are input into CRP_Net_x for further processing. First, an RCU unit is used to fine-tune the weight of ResNet training. Then, the MRF module is utilized to fuse the input features from ResNet and the output features from lower resolutions. Moreover, the CRP module is employed to extract global semantic information, and finally, the result is output from an RCU.

The dense connection is mainly reflected in the connection between the features of different resolutions in the decoding network. As shown in Figure 5, the input of CRP_Net_x ($x = 1, 2, 3$) contains two parts: one is the input of the feature map from residual network, and the other is the feature map from all CRP_Net_x with lower resolutions. This allows each CRP_Net Block to make full use of the previous middle and high-level semantic features, and finally input them into the DAFE module, thus repeatedly fusing and re-extracting the features. The dense connection fuses the features of four resolutions, which makes the training gradient transfer effectively between the CRP_Net module and the DAFE module, and avoids the disappearance of the gradient.

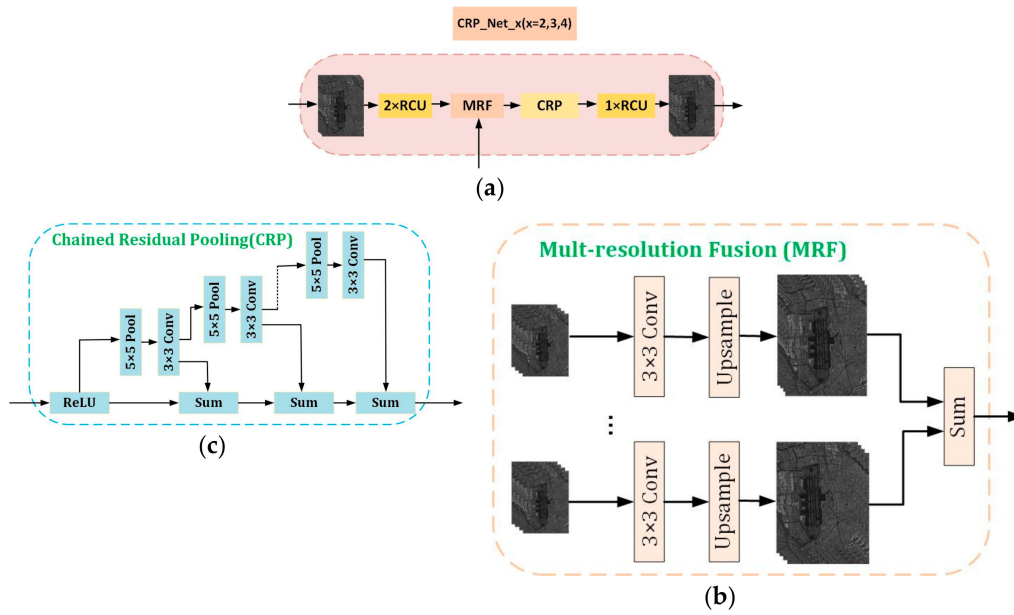


Figure 6. The internal structure of CRP_Net_x. (a) is the overall structure of CRP_Net_x; (b) is the MRF structure; (c) is the Chained Residual Pooling (CRP) structure.

3.4.2. Dual Attention Mechanism

In this paper, dual attention is incurred with the highest-resolution features produced by ResNet to form a DAEF module, while the other three CRP_Net_x ($x = 1, 2, 3$) with lower resolutions remain unchanged. The input of the DAEF module includes the highest resolution features from ResNet and all of the high-level semantic features extracted from CRP_Net_x (1, 2, 3). These features are fused by the MRF module. Then, the PAM and CAM are employed to weigh the position features and channel features. Moreover, the weighted features are fused by the MRF module and perform a RCU operation. Finally, the extraction results are generated by the Softmax function. The detailed implementation process of the PAM and CAM in this paper is shown in Figure 7.

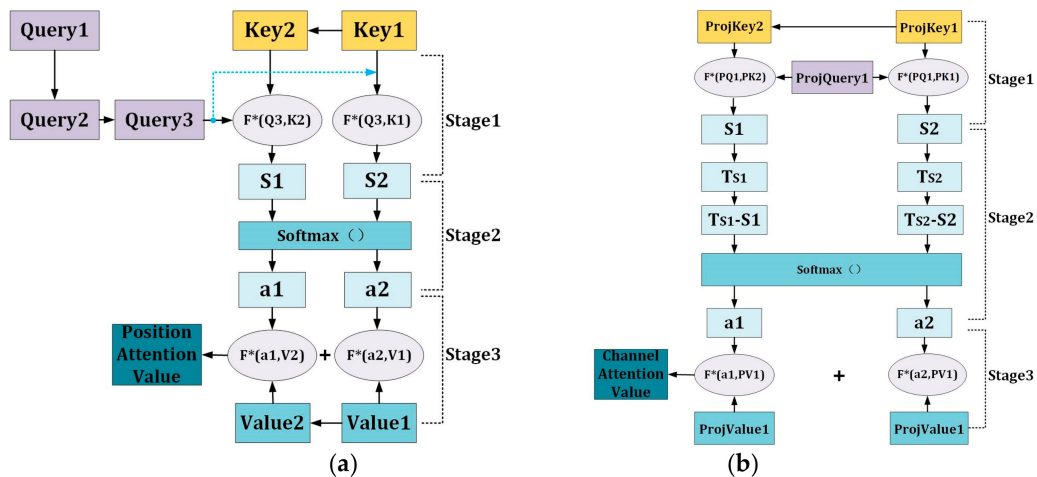


Figure 7. The detailed implementation process of Position Attention Mechanism (PAM) and Channel Attention Mechanism (CAM). (a) PAM; (b) CAM.

• The implementation of PAM

As shown in Figure 7a, the detailed implementation process of the PAM can be divided into three stages. Query1, Key1, and Value1 are all the position variables generated by the input. Query2 is

obtained when the ‘reshape’ operation is performed on Query1, and Query3 is acquired when Query2 is transposed. Reshaping Key1 and Value1, we can gain Key2 and Value2, respectively. All of the element values of the input image can be regarded as a collection of <Query, Key>. In the first stage, a matrix multiplication function is introduced to calculate the similarity of the positional relationship between the two variables.

$$\begin{aligned} S1 &= \text{Similarity}(\text{Query3}, \text{Key2}) = \text{Query3} * \text{Key2} \\ S2 &= \text{Similarity}(\text{Query3}, \text{Key1}) = \text{Query3} * \text{Key1} \end{aligned} \quad (9)$$

In the second stage, softmax is introduced to numerically convert the S1 and S2 obtained in the first stage. One purpose is to perform normalization, and the other purpose is to emphasize the weight of elements in important positions, which is more prominent through the internal mechanism of softmax.

$$\begin{aligned} a1 &= \text{Softmax}(S1) = \frac{e^{S1}}{e^{S1} + e^{S2}} \\ a2 &= \text{Softmax}(S2) = \frac{e^{S2}}{e^{S1} + e^{S2}} \end{aligned} \quad (10)$$

The calculated a1 and a2 are the weight coefficients corresponding to Value2 and Value1. In addition, matrix multiplication is performed, and then the position attention values are produced after the operation of the weight and sum.

$$\text{Position Attention} = a1 * \text{Value2} + a2 * \text{Value1} \quad (11)$$

Through the above calculation of the three stages, the position attention value for Query3 can be obtained.

- **The implementation of CAM**

The calculation process for the CAM is shown in Figure 7b. Unlike the PAM, ProjQuery1, ProjKey1 and ProjValue1 are all directly reshaped from the input, and ProjKey2 is generated after transposing ProjKey1. The calculation method of S1 and S2 is the same as that of PAM, but the difference is that S1 and S2—obtained by CAM in the first stage—are not directly input to softmax. In the CAM, the maximum value of elements in each dimension of the channel tensor is selected, and the dimension is expanded. Moreover, the total number of elements from the matrix is subtracted from the total number of elements after the expansion.

$$\begin{aligned} T_{S1} &= \text{Expand}_{\dim(\text{Max}(S1))} \\ T_{S2} &= \text{Expand}_{\dim(\text{Max}(S2))} \end{aligned} \quad (12)$$

In the second stage, softmax is introduced to perform numerical conversion on $T_{S1} - S1$ and $T_{S2} - S2$ obtained in the first stage.

$$\begin{aligned} a1 &= \text{Softmax}(T_{S1} - S1) = \frac{e^{T_{S1} - S1}}{e^{T_{S1} - S1} + e^{T_{S2} - S2}} \\ a2 &= \text{Softmax}(T_{S2} - S2) = \frac{e^{T_{S2} - S2}}{e^{T_{S1} - S1} + e^{T_{S2} - S2}} \end{aligned} \quad (13)$$

The calculated values of a1 and a2 are respectively matrix-multiplied with ProjValue1, and then weighted and summed to obtain the value of the channel attention value.

$$\text{Channel Attention} = (a1 + a2) * \text{ProjValue1} \quad (14)$$

The PAM weights the position features of all semantic features, and selectively aggregates features at each position. Regardless of whether the position is near or far, similar features are related to each other. The CAM integrates the relationships between all feature channels, and selectively emphasizes the interdependent channel features. The entire dual attention weights and selects the position features

and channel features, retains useful features, discards low-level features, and further improves features representation to make the segmentation results more precise.

3.5. The Training Process of the Framework

The framework of the MDDA network proposed in this paper consists of two parts: one is the encoding network ResNet_101, and the other is the decoding network including the CRP_Net_x ($x = 1, 2, 3$) module and DAFE module. In this paper, dense connections were utilized between the CRP_Net_x (1, 2, 3) modules and also between the CRP_Net module and the DAFE module. In the DAFE module, the dual attention mechanism is introduced. The entire training process of the MDDA network is as follows:

Input: Datasets including small SAR images and corresponding ground truth.

Training:

- (1) Initializing of input data: the coding network loads training data from the ImageNet pre-trained model.
- (2) The loaded training data are input to ResNet-101 to extract multi-level features.
- (3) The decoding network fuses and re-extracts the features extracted by the coding network. Of which, dense connections enhance gradient propagation between features, and dual-attention selects the features by weights.
- (4) Back propagation (BP) algorithm performs end-to-end training for the whole network.
- (5) The softmax function calculates the probabilities that the network output is mapped to the runway and background categories by the following formula.

$$\hat{p}^k = \frac{e^{X^k}}{\sum_{k=1}^K e^{X^k}}$$

where X^k represents the number of pixels corresponding to the k -th category, and K is the number of the sample categories. \hat{p}^k denotes the probability of the k -th category being predicted correctly after the softmax function.

The network employs Cross Entropy Loss as the optimization function, which is shown as follows:

$$E = \sum_{k=1}^K p^k \log(\hat{p}^k)$$

where p^k is defined as a variable 0 or 1, that is, when the predicted category is the same as the sample category, $p^k = 1$; otherwise, $p^k = 0$.

Since there were only two types of targets in this paper, runway areas and background, we can directly use the binary classification of the cross-entropy loss function.

$$E = p^k \log(\hat{p}^k) + (1 - p^k) \log(1 - \hat{p}^k)$$

Output: Trained model for airport extraction.

4. Experiment and Results

4.1. Dataset Used in the Experiment

To validate the proposed framework MDDA in this paper, SAR images with 1-m resolution from Gaofen-3 system were utilized. Many large-scale SAR images including airports were used in the experiment. First, SAR images were down-sampled by five times to generate medium resolution images. Then, the ground truth was produced by Image Labeler of MATLAB, which includes runway

areas and the background. The runway area, which includes runways, taxiways, and parking lots, is marked red and other targets are regarded as background. In addition, the down-sampled SAR images and corresponding ground truth were cut into small images with 500×500 pixels to generate the dataset. After data augmentation using flip, mirror, and shift, a total of 2479 samples was achieved, and the ratio of the training set to the validation set was 3:1. To test the model generated by training the proposed framework, four SAR images including airports unused in making the dataset were utilized to extract runway areas of airports.

Figure 8 shows some samples of the airports in the experiment. Figure 8a–c are SAR images, the ground truth, and the optical remote sensing image of Hongqiao Airport in Shanghai, China. Figure 8d–f denote the three images of Capital airport, the same as Hongqiao Airport. From these samples, we can note the background of Hongqiao Airport is relatively simple, while the background of Capital Airport is more complicated and harder to extract.

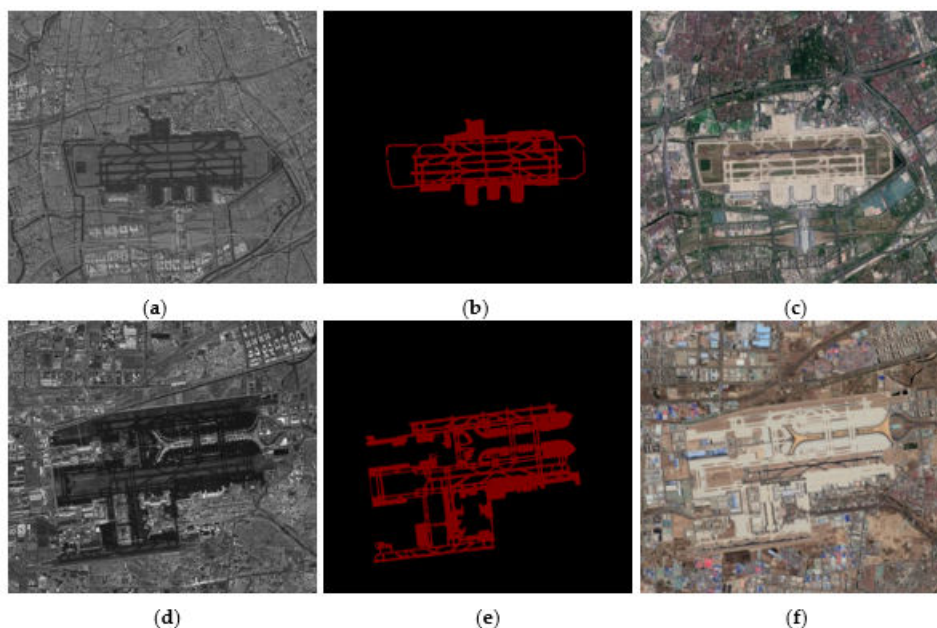


Figure 8. Airport images and corresponding ground truth. (a–c) denote the SAR image, ground truth, and the corresponding optical remote sensing image for Hongqiao Airport of China. (d–f) are the SAR image, ground truth, and optical remote sensing image for Capital Airport.

4.2. Evaluation Measurements

To evaluate the extraction precision of airports, pixel accuracy (PA) and intersection over union (IoU) were used following previous works [21–23,34]. PA denotes the ratio of correctly extracted pixels to the total pixels of the type of targets, and IoU represents the ratio of intersection and union of extracted results to the ground truth. Mean pixel accuracy (MPA) is the mean proportion of correctly classified pixels for all categories, and mean intersection over union (MIoU) denotes the mean IoU of all types of targets. The specific calculation formulas are as follows [34].

$$PA = \frac{\sum_{i=0}^k P_{ii}}{\sum_{i=0}^k \sum_{j=0}^k P_{ij}} \quad (15)$$

$$MPA = \frac{1}{k+1} \frac{\sum_{i=0}^k P_{ii}}{\sum_{i=0}^k \sum_{j=0}^k P_{ij}} \quad (16)$$

$$IoU = \sum_{i=0}^k \frac{P_{ii}}{\sum_{j=0}^k P_{ij} + \sum_{j=0}^k P_{ji} - P_{ij}} \quad (17)$$

$$MIoU = \frac{1}{k+1} \sum_{i=0}^k \frac{P_{ii}}{\sum_{j=0}^k P_{ij} + \sum_{j=0}^k P_{ji} - P_{ij}} \quad (18)$$

where $k + 1$ is the total number of categories (because the background is also a category). P_{ij} denotes the number of pixels that originally belong to class i but are predicted to be class j , which are false positive samples. P_{ji} indicates the number of pixels that originally belong to class j , but are predicted to be class i , which are false negative samples. P_{ii} means the number of pixels correctly classified in class i .

4.3. Experiment Analysis and Evaluation

To test the proposed framework of MDDA, four SAR images covering airports unused in training and validation were utilized. Furthermore, two popular deep neural networks for semantic segmentation (RefineNet [17] and DeepLabV3 [21]) were used as reference studies. DeepLabV3 was an excellent network for semantic segmentation, which achieved the best performance on the PASCAL VOC 2012 with other state-of-art models in 2017. RefineNet presented a multi-level structure and chained residual pooling strategy to accomplish semantic segmentation, which also attained much better performance than the vast majority of the networks at that time in PASCAL VOC 2012.

• The extraction result of Airport I

Figure 9 indicates the extraction results for Airport I. Figure 9a is the SAR image of Airport I from the Gaofen-3 system with a 1-m resolution and Figure 9b is the down-sampled image by five times of (a). From what we can see, the texture of the targets in (a) is much clearer than that in (b). Figure 9c is the ground truth of the airport corresponding to (b). Figure 9d–f are the extraction results of the airport generated by RefineNet, DeepLabV3, and the proposed framework MDDA, respectively. Figure 9g–i denote the fusion maps of (d), (e), and (f) with (b), respectively. Figure 9j–l represent the fusion maps of Figure 9a and the corresponding up-sampled extraction results of Figure 9d–f by five times.

Airport I belongs to the civil airport. According to Figure 9a, there are many buildings around Airport I, and the traffic lines are intertwined and complicated. The airport has a relatively obvious characteristic difference from the surrounding ground features, visually showing a large area of gray and black, and the runway area is black. Comparing Figure 9d–f with the ground truth in Figure 9c, we can see that result (f) had the highest overlap with (c), which indicates that MDDA extracted the best result for Airport I. From Figure 9g, RefineNet had a great number of missed detections, most of which were parking lots and some runways. Figure 9h shows that DeepLabV3 had some missed detections and false detections, and the integrity of the edge extraction for the runway area was not high. While for MDDA, only a small part of the runway was not detected, the missed detection rate was low, and there was no false detection. Comparing Figure 9g–i, we can see much more detailed information in Figure 9j–l because of the high resolution, which also demonstrates that we can obtain a satisfactory extraction result of the airport for SAR images with high resolution by the proposed MDDA framework.

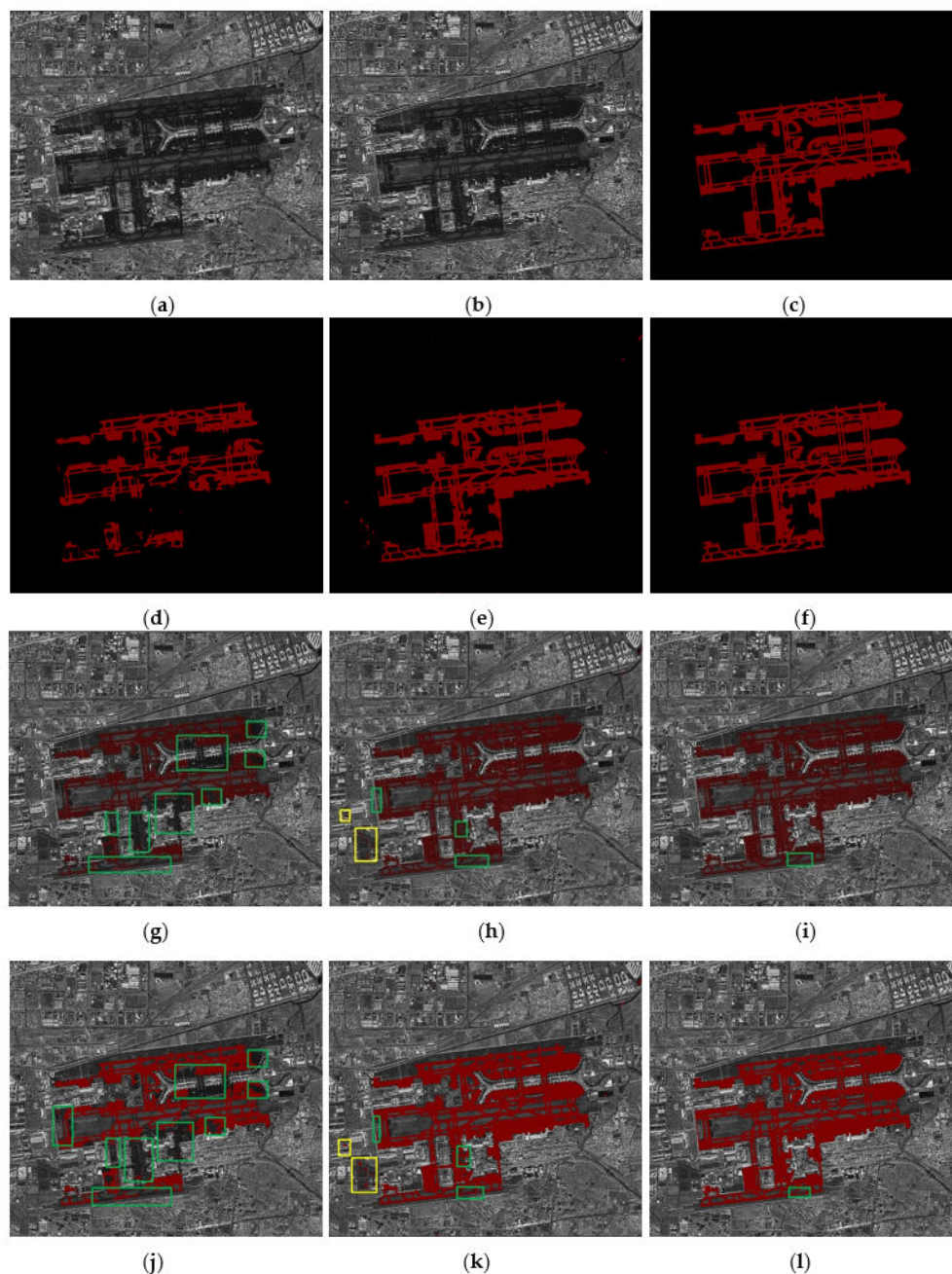


Figure 9. The experiment result for Airport I. (a) SAR image of Airport I from Gaofen-3. (b) The down-sampled SAR image of (a) by 5 times. (c) The ground truth of the airport for (b). (d) The extraction result of (b) by RefineNet. (e) The extraction result of (b) by DeepLabV3. (f) The extraction result of (b) by MDDA. (g) The fusion map of (d,b). (h) The fusion map of (e,b). (i) The fusion map of (f,b). (j) The fusion map of (a,d) up-sampled by 5 times. (k) The fusion map of (a,e) up-sampled by 5 times. (l) The fusion map of (a,f) up-sampled by 5 times.

- **The result of Airport II**

Figure 10 demonstrates the extraction results for Airport II. Figure 10a–l are the same type of images as the corresponding images in Figure 9. According to Figure 10a, the buildings around the airport are sparse, but there is a large area of water, and the characteristics of the water and the runway area are very similar. As shown in Figure 10h,k, DeepLabV3 mis-detected a large number of water bodies as runway areas, and there were also some missed detections for the airport. From Figure 10g,j, we found that RefineNet acquired a better extraction result than DeepLabV3, but there were still some

false alarms and missed detections, while according to Figure 10f,i,l, MDDA achieved the best detection performance for Airport II. There was no false alarm and only a few missed detections.

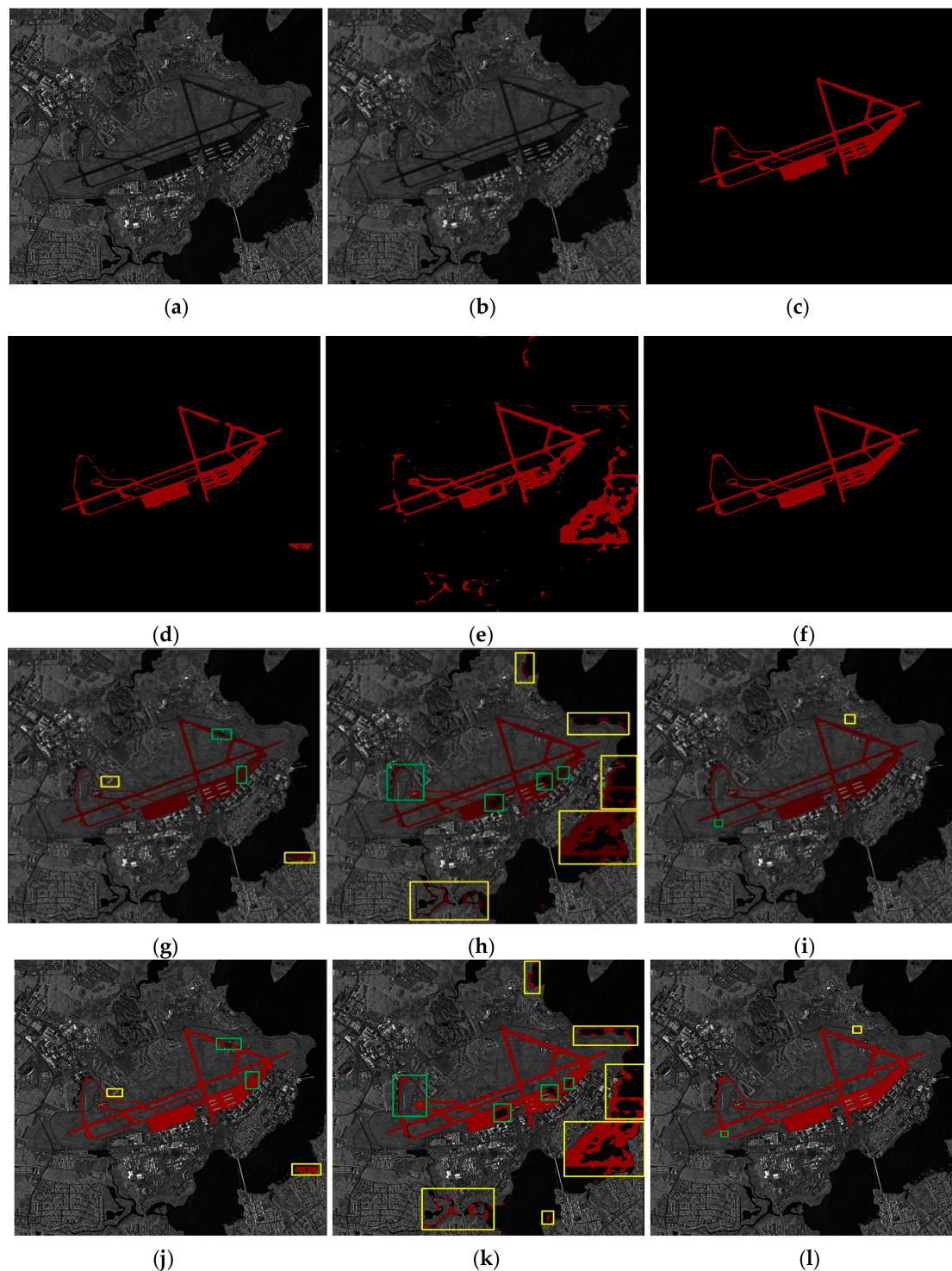


Figure 10. The experiment result for Airport II. (a) SAR image of Airport II from Gaofen-3. (b) The down-sampled SAR image of (a) by 5 times. (c) The ground truth of the airport for (b). (d) The extraction result of (b) by RefineNet. (e) The extraction result of (b) by DeepLabV3. (f) The extraction result of (b) by MDDA. (g) The fusion map of (d,b). (h) The fusion map of (e,b). (i) The fusion map of (f,b). (j) The fusion map of (a,d) up-sampled by 5 times. (k) The fusion map of (a,e) up-sampled by 5 times. (l) The fusion map of (a,f) up-sampled by 5 times.

- **The result of Airport III**

Figure 11 indicates the extraction results for Airport III. There are dense buildings and terraces to the west of the airport, and there are also waters nearby. Figure 11a–l are also the same type of images as the corresponding images in Figure 9.

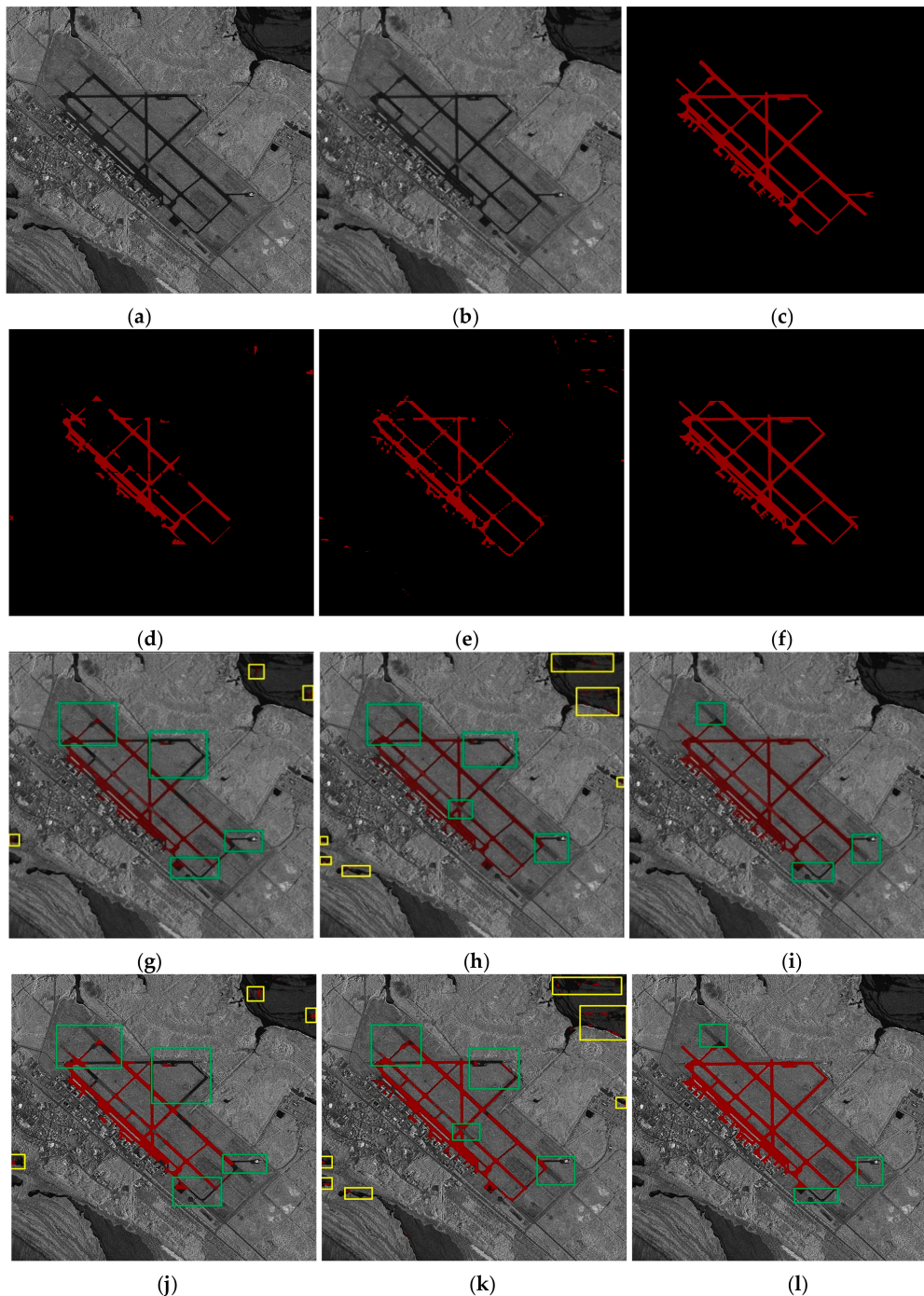


Figure 11. The experiment result for Airport III. (a) SAR image of Airport III from Gaofen-3. (b) The down-sampled SAR image of (a) by 5 times. (c) The ground truth of the airport for (b). (d) The extraction result of (b) by RefineNet. (e) The extraction result of (b) by DeepLabV3. (f) The extraction result of (b) by MDDA. (g) The fusion map of (d,b). (h) The fusion map of (e,b). (i) The fusion map of (f,b). (j) The fusion map of (a,d) up-sampled by 5 times. (k) The fusion map of (a,e) up-sampled by 5 times. (l) The fusion map of (a,f) up-sampled by 5 times.

According to the comprehensive result images (d), (e), (f) and fusion maps (g), (h), (i), RefineNet and DeepLabV3 both misdetected the water areas as runway areas (such as yellow boxes), and there were many missed detections (such as green boxes). Due to the obvious visual difference between the extended runway at both ends of the airport runway and the main runway area, none of the three networks detected the extended runway. Except for the missed extended runway areas, the rest of the runway areas were all detected by MDDA, and there were no false alarms.

- **The result of Airport IV**

Figure 12 presents the extracted results of Airport IV, which is Hongqiao Airport. There are considerable road networks, which are very likely to cause false detections. Figure 12a–l are also the same type of images as the corresponding images in Figure 11.

Based on the results shown in Figure 12, all three networks can extract airport runways, and the crossing roads have not become false alarms. According to Figure 12d,g,j, we can see that RefineNet missed a great many runways (marked by green boxes), which caused the relatively worst detection performance of the three networks. DeepLab V3 has greatly reduced the runway areas missed by RefineNet, but there were some false alarms (marked by yellow boxes), and the overall detection performance was highly improved. According to Figure 12f,i,l, there were the least missed runways and no false alarms in the results generated by the MDDA network, so the extracted results were the best of the three networks for runways of the airport from SAR images.

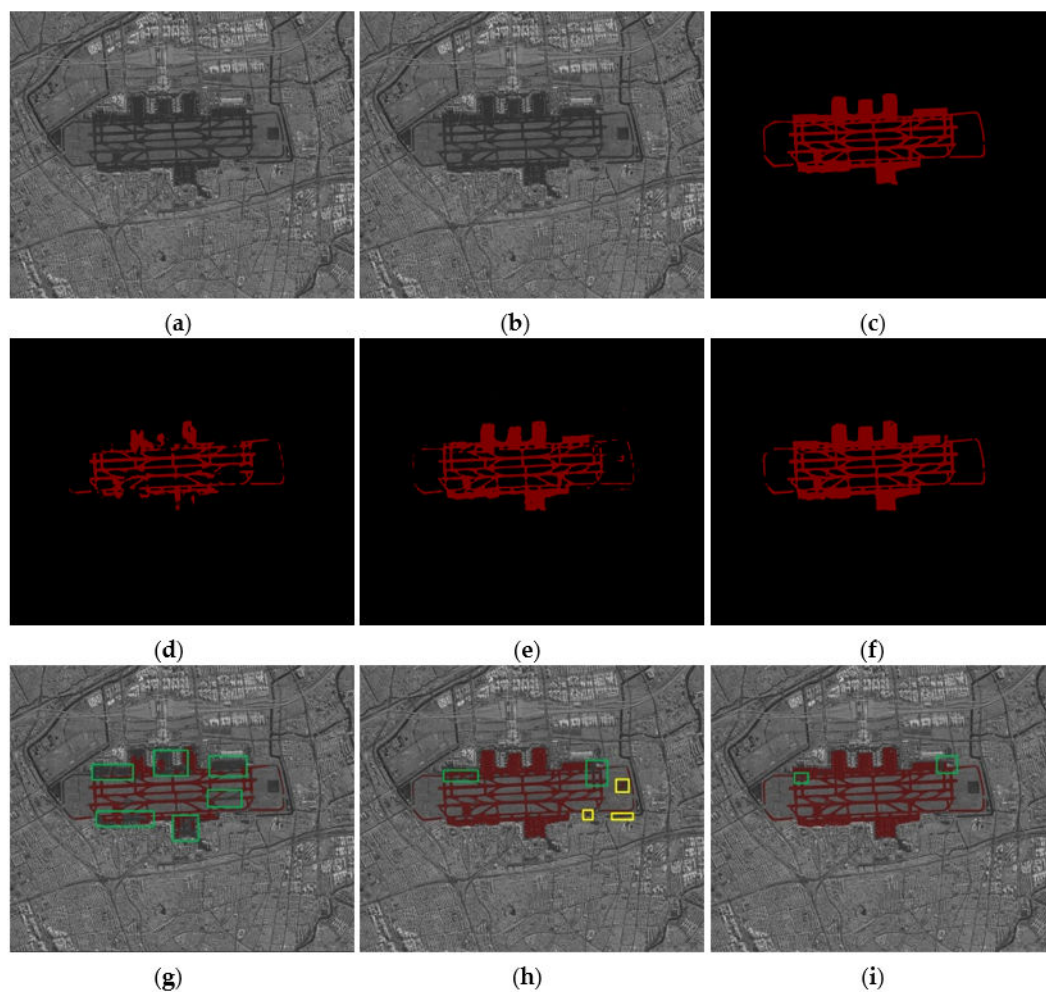


Figure 12. Cont.

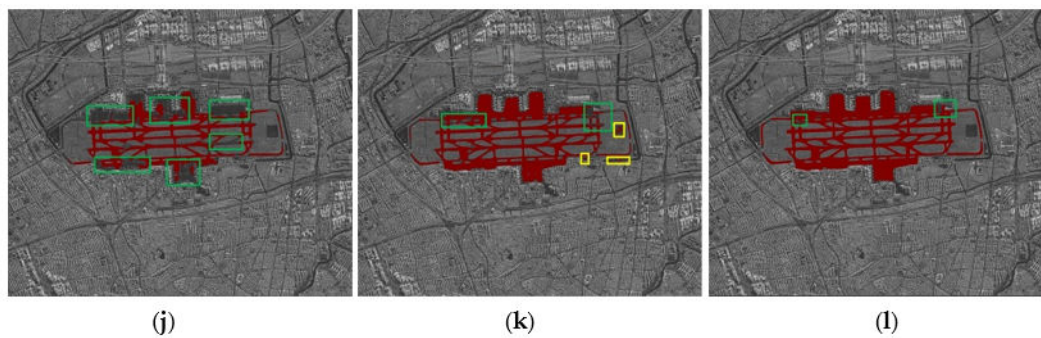


Figure 12. The experiment result for Airport IV. (a) SAR image of Airport IV from Gaofen-3. (b) The down-sampled SAR image of (a) by 5 times. (c) The ground truth of the airport for (b). (d) The extraction result of (b) by RefineNet. (e) The extraction result of (b) by DeepLabV3. (f) The extraction result of (b) by MDDA. (g) The fusion map of (d,b). (h) The fusion map of (e,b). (i) The fusion map of (f,b). (j) The fusion map of (a,d) up-sampled by 5 times. (k) The fusion map of (a,e) up-sampled by 5 times. (l) The fusion map of (a,f) up-sampled by 5 times.

To analyze the extraction performance of the airports, Table 1 gives the extraction accuracies of different networks for the four airports. According to Table 1, the proposed MDDA framework had the best extraction performance of airports, which had the least number of missed detections and almost no false alarms. The mean pixel accuracy (MPA) of runway areas reached 0.9811 and the MIoU reached 0.9707, which proved the superiority of the MDDA.

Table 1. The extraction accuracy for airports by different networks.

Network	Airports	Runway Areas		Background		MPA	MIoU
		PA	IoU	PA	IoU		
RefineNet	Airport I	0.6386	0.6306	0.9980	0.9447	0.8188	0.7877
	Airport II	0.8995	0.8552	0.9977	0.9932	0.9486	0.9242
	Airport III	0.6062	0.5957	0.9990	0.9772	0.8026	0.7865
	Airport IV	0.6024	0.5946	0.9990	0.9698	0.8007	0.7822
	Mean					0.8427	0.8202
DeepLabV3	Airport I	0.9452	0.8891	0.9901	0.9817	0.9677	0.9354
	Airport II	0.8875	0.4619	0.9588	0.9540	0.9232	0.7228
	Airport III	0.6689	0.6411	0.9975	0.9792	0.8332	0.8102
	Airport IV	0.8288	0.8166	0.9989	0.9861	0.9139	0.9014
	Mean					0.9095	0.8425
MDDA Net	Airport I	0.9849	0.9706	0.9977	0.9953	0.9913	0.9830
	Airport II	0.9845	0.9609	0.9989	0.9982	0.9917	0.9796
	Airport III	0.9189	0.9016	0.9989	0.9943	0.9589	0.9480
	Airport IV	0.9664	0.9486	0.9986	0.9960	0.9825	0.9723
	Mean					0.9811	0.9707

RefineNet presented the most missed detection results. There were large areas of runway areas that were not detected in three airports, except for the second airport, and there were a small number of false detection results in Airport II, Airport III, and Airport IV. DeepLabV3 had the most false-alarms. For Airport II, the false alarm was the most serious, and they were all false alarms in the extraction results generated by three networks. The experimental results showed that RefineNet and DeepLabV3 do not have the ability to learn airport features and cannot distinguish runway areas from similar areas, resulting in poor detection integrity and false alarms. While for MDDA, the transmission between features is enhanced by introducing dense connection, and redundant features were abandoned and useful features were retained via incurring the dual attention mechanism. Therefore, the network's ability to learn features is improved, which makes the runway extraction results free of false alarms

and high extraction completeness. Compared with the other two networks, MDDA could almost completely extract the entire runway edge line.

In order to more clearly analyze the detailed information of the extracted airport, Figure 13 shows an enlarged view of a small area of Airport I. Figure 13a–l are also the same type of images as the corresponding images in Figure 12. It can be seen from the views of (g), (h), and (i) that MDDA can extract the details and edge information well. RefineNet is a typical semantic segmentation network, but the decoding network only uses simply transferring features one by one, so the extraction effect was poor as we could see many missed detections (as shown in the green boxes). DeepLabV3 adds hole convolution to expand the receptive field, but the lack of attention mechanism makes the feature learning redundant. Therefore, it cannot extract the detailed information well, and is prone to false alarms. The proposed MDDA network in this paper improved these problems, and we can see from the detailed images that MDDA could extract the airport runway area much better. In addition, we can see more detailed information in Figure 13j–l than in Figure 13g–i due to their high resolutions.

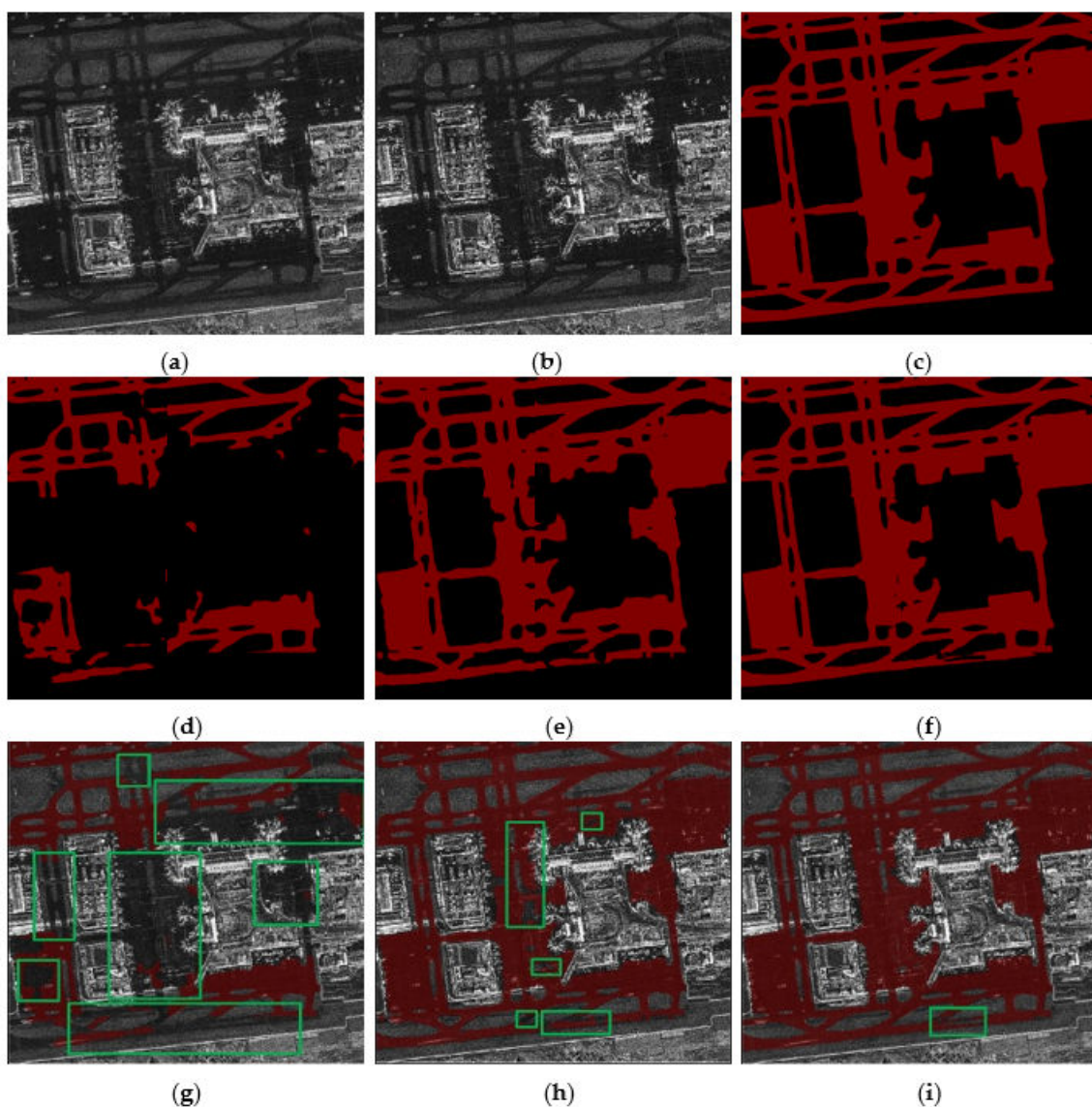


Figure 13. Cont.

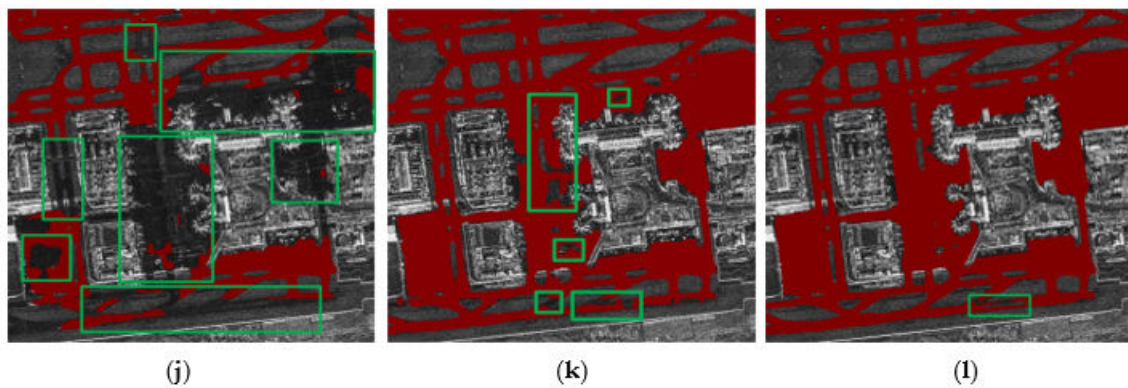


Figure 13. The enlarged view of a small part of Airport I. (a) SAR image of a part of Airport I from Gaofen-3. (b) The down-sampled SAR image of (a) by 5 times. (c) The ground truth of the airport for (b). (d) The extraction result of (b) by RefineNet. (e) The extraction result of (b) by DeepLabV3. (f) The extraction result of (b) by MDDA. (g) The fusion map of (d,b). (h) The fusion map of (e,b). (i) The fusion map of (f,b). (j) The fusion map of (a,d) up-sampled by 5 times. (k) The fusion map of (a) and (e) up-sampled by 5 times. (l) The fusion map of (a,f) up-sampled by 5 times.

In order to better verify the performance of the proposed network, more images were used for testing based on data augmentation techniques [34]. Due to the limited large-scale airport images of high-resolution SAR images, the four airports tested in this paper were horizontally flipped, vertically flipped, rotated 90° clockwise, and rotated 90° counterclockwise to obtain 16 new airport images. Then, we utilized three networks to test the 16 images to acquire the extracted accuracy for airports, respectively. For the four images of each airport, we could obtain a mean accuracy for every network, which is shown in Table 2. Compared with Table 1, we found that the accuracy of each network for every airport was nearly the same, which illustrates the stability of the three networks for extracting the runways of the airports.

Table 2. The extraction accuracy for augmented airports by different networks.

Network	Airports	Runway Areas		Background		MPA	MIoU
		PA	IoU	PA	IoU		
RefineNet	Airport I	0.6384	0.6305	0.9982	0.9448	0.8189	0.7878
	Airport II	0.8998	0.8555	0.9978	0.9933	0.9488	0.9245
	Airport III	0.6058	0.5955	0.9988	0.9770	0.8022	0.7860
	Airport IV	0.6029	0.5948	0.9993	0.9700	0.8010	0.7826
	Mean					0.8427	0.8200
DeepLabV3	Airport I	0.9458	0.8896	0.9906	0.9822	0.9681	0.9359
	Airport II	0.8879	0.4622	0.9594	0.9545	0.9237	0.7232
	Airport III	0.6695	0.6415	0.9981	0.9797	0.8338	0.8107
	Airport IV	0.8286	0.8165	0.9986	0.9860	0.9137	0.9013
	Mean					0.9098	0.8428
MDDA Net	Airport I	0.9855	0.9709	0.9981	0.9956	0.9916	0.9833
	Airport II	0.9844	0.9608	0.9990	0.9982	0.9918	0.9796
	Airport III	0.9187	0.9015	0.9991	0.9945	0.9590	0.9482
	Airport IV	0.9665	0.9487	0.9988	0.9961	0.9827	0.9724
	Mean					0.9813	0.9709

Figure 14 demonstrates the extraction results for one image of the augmented 16 airport images, which is the horizontal flipped image of Airport I. Compared with Figure 9, we found that they were nearly the same detection accuracy according to Table 2 and Figure 14, which also demonstrates the stability of the networks.

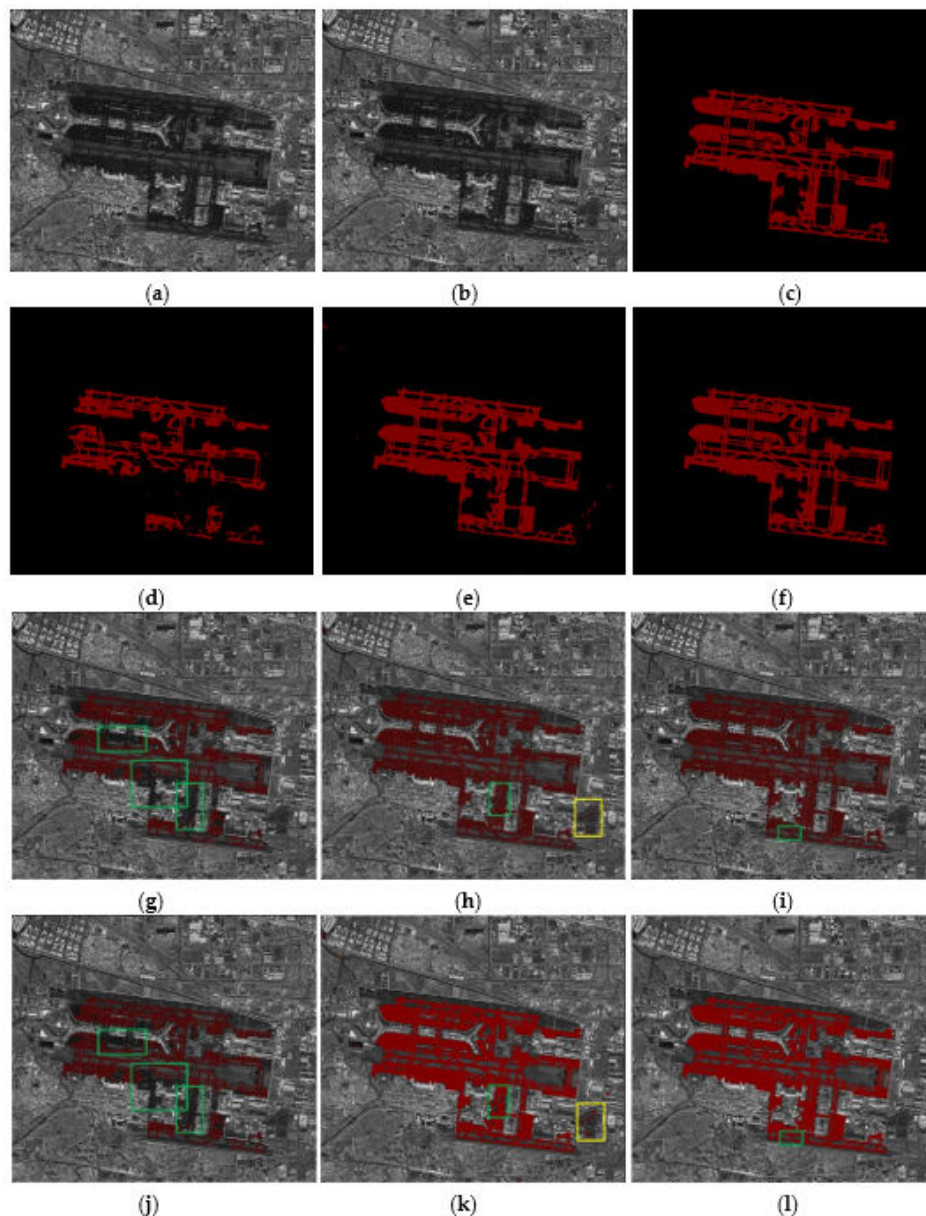


Figure 14. The experimental results for the horizontal flipped Airport I. (a) SAR image from Gaofen-3. (b) The down-sampled SAR image of (a) by 5 times. (c) The ground truth of the airport for (b). (d) The extraction result of (b) by RefineNet. (e) The extraction result of (b) by DeepLabV3. (f) The extraction result of (b) by MDDA. (g) The fusion map of (d,b). (h) The fusion map of (e,b). (i) The fusion map of (f,b). (j) The fusion map of (a,d) up-sampled by 5 times. (k) The fusion map of (a) and (e) up-sampled by 5 times. (l) The fusion map of (a,f) up-sampled by 5 times.

5. Discussion

In this paper, we propose a multi-level and densely dual attention (MDDA) network to extract the runways of the airport from SAR images with high-resolution. First, the high-resolution SAR images were down-sampled to generate medium resolution SAR images, and then the samples were produced. Second, the MDDA network was utilized to extract the runways of the airport by making full use of the effective features of the runway areas, where ResNet, the dual-attention mechanism, dense connection, and multi-level structure were integrated. Finally, the bilinear interpolation was incurred to achieve the extraction results of the runways for the high-resolution SAR images. According to the results of

the experiments, we noted that MDDA could acquire a satisfactory performance for runway extraction of the airport and obtained the highest accuracy of the three networks.

In addition, we also noted that the training speed of MDDA was relatively slow, which will be our next research direction. Moreover, only SAR images from the Gaofen-3 system were utilized in the experiment, so high-resolution SAR images with different bands and different resolutions will be tested in our further research. Once the runways are extracted, the aircrafts in the airport can be detected more accurately, which not only reduces the false alarms of aircraft detection, but also remarkably increases the detection speed. Therefore, this is also our future work.

6. Conclusions

To accomplish the automatic airport detection from high-resolution SAR images, a new framework named MDDA was proposed, which integrated ResNet, dense connection, CRP, and the dual attention mechanism. The dense connection takes the advantage of features generated by ResNet at different levels. The dual attention mechanism extracts the position features and channel features respectively, and weights them with different values according to their significance to classification. To implement airport detection from SAR images with high-resolution, two additional processes are performed. One is down-sampling the original SAR images to the medium resolution ones, so that the samples can contain more spatial features of the airport. The other is up-sampling the extraction result generated by the MDDA network to achieve the airport extraction of SAR images with the same resolution.

Three Gaofen-3 SAR images including different airports were utilized to test the proposed MDDA framework. Compared with two existing semantic segmentation networks, namely, RefineNet and DeepLabV3, MDDA achieved much better performance for airport extraction, which reached 0.98 in MPA and 0.97 in MIoU. In addition, it can also be seen from the extraction results that there were few missed detection areas and no false alarms for MDDA, which indicates that it can effectively extract the airport runway areas, and the integrity of the details remains outstanding.

Author Contributions: L.C. and S.T. proposed the new network, designed the experiments, and produced the results; S.T. made the SAR dataset; S.T., L.C., Z.P., J.X., Z.Y., X.X., and P.Z. contributed to the discussion of the results. L.C. and S.T. drafted the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partly supported by the National Natural Science Foundation (No. 41201468, 41701536, 61701047, 41674040) of China, partly funded by the Natural Science Foundation of Hunan Province (No. 2017JJ3322, 2019JJ50639), and partly funded by the Foundation of Hunan, Education Committee, under Grants No. 16B004 and No. 16C0043.

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

References

1. Ferretti, A.; MontiGuarnieri, A. InSAR Principles-Guidelines for SAR Interferometry Processing and Interpretation. *J. Financ. Stabil.* **2007**, *19*, 156–162. [[CrossRef](#)]
2. Zhang, L.P.; Zhang, H. A Fast Method of Airport Detection in Large 2 scale SAR Image with High Resolution. *J. Image Gr.* **2010**, *15*, 1112–1120. [[CrossRef](#)]
3. Yan, J.; Xu, J.; Ai, S.; Li, D.; Wang, Z. Airport runway detection algorithm based on local multi-features. *Chin. J. Sci. Instrum.* **2014**, *35*, 1714–1720.
4. Zhu, D.; Wang, B. Airport detection based on near parallelity of line segments and GBVS saliency. *J. Infrared Millim. Waves* **2015**, *34*, 375–384. [[CrossRef](#)]
5. Kou, Z.; Shi, Z.; Liu, L. Airport detection based on Line Segment Detector. In Proceedings of the International Conference on Computer Vision in Remote Sensing, Xiamen, China, 16 December 2012; pp. 1–6.
6. 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 Global Symposium on Millimeter-Waves, Harbin, China, 27–30 May 2012; pp. 462–465.
7. Zhu, D.; Wang, B. Airport Target Detection in Remote Sensing Images: A New Method Based on Two-Way Saliency. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 1096–1100. [[CrossRef](#)]

8. Liu, N.; Cui, Z. Airport Detection in Large-Scale SAR Images via Line Segment Grouping and Saliency Analysis. *IEEE Geosci. Remote Sens.* **2018**, *15*, 434–438. [[CrossRef](#)]
9. Zhang, L.; Zhang, H. A method of Airport Extraction Based on Template searching from High Resolution SAR Image. *Remote Sens. Inf.* **2010**, *2*, 30–35. [[CrossRef](#)]
10. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [[CrossRef](#)]
11. Chang, L.; Deng, X.M. Convolutional Neural Networks in Image Understanding. *Acta Autom. Sin.* **2016**, *42*, 1300–1312. [[CrossRef](#)]
12. Jonathan, L.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
13. Tang, H.; He, C. SAR image scele classification with fully convolutional network and modified conditional random field-recurrent neural network. *J. Comput. Appl.* **2016**, *36*, 3436–3441. [[CrossRef](#)]
14. Badrinarayanan, V.; Kendall, A. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Scene Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [[CrossRef](#)]
15. Chen, L.C.; George, P. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. *arXiv* **2014**, arXiv:1412.7062v4.
16. Chen, L.C.; George, P. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *40*, 834–848. [[CrossRef](#)] [[PubMed](#)]
17. Lin, G.; Anton, M.; Shen, C. RefineNet: Multi-path Refinement Networks for High-Resolution Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1925–1934.
18. Peng, C.; Zhang, X.; Yu, G. Large Kernel Matters—Improve Semantic Segmentation by Global Convolutional Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1743–1751.
19. Zhang, Z.; Zhang, X.; Peng, C. ExFuse: Enhancing Feature Fusion for Semantic Segmentation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 273–288.
20. Zhang, H.; Shi, J.; Qi, X. Pyramid Scene Parsing Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6230–6239.
21. Liang-Chieh, C.; Papandreou, G. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:1706.05587v3.
22. Chen, L.; George, P. Encoder-Decoder with Atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 801–818.
23. Liu, C.; Chen, L.C.; Schroff, F. Auto-DeepLab: Hierarchical Neural Architecture Search for Semantic Image Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 82–92.
24. Yu, D.; Zhang, N. Airport detection using convolutional neural network and salient feature. *Bull. Surv. Mapp.* **2019**, *7*, 44–49. [[CrossRef](#)]
25. Xiao, Z.; Gong, Y. Airport Detection Based on a Multiscale Fusion Feature for Optical Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1469–1473. [[CrossRef](#)]
26. Zeng, F.; Cheng, L. A Hierarchical Airport Detection Method Using Spatial Analysis and Deep Learning. *Remote Sens.* **2019**, *11*, 2204. [[CrossRef](#)]
27. Li, S.; Xu, Y. Remote Sensing Airport Detection Based on End-to-End Deep Transferable Convolutional Neural Networks. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1640–1644. [[CrossRef](#)]
28. He, K.; Zhang, X.; Ren, S. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
29. He, K.; Zhang, X.; Ren, S. Identity Mappings in Deep Residual Networks. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 630–645.
30. Huang, G.; Liu, Z.; Laurens, V.D.M. Densely Connected Convolutional Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.

31. Zhang, P.; Chen, L. Automatic Extraction of Water and Shadow from SAR Images Based on a Multi-Resolution Dense Encoder and Decoder Network. *Sensors* **2019**, *19*, 3576. [[CrossRef](#)]
32. Chen, L.; Cui, X. A New Deep Learning Algorithm for SAR Scene Classification Based on Spatial Statistical Modeling and Features Re-Calibration. *Sensors* **2019**, *19*, 2479. [[CrossRef](#)]
33. Fu, J.; Liu, J.; Tian, H. Dual Attention Network for Scene Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 3146–3154.
34. Garcia-Garcia, A.; Orts-Escolano, S. A Review on Deep Learning Techniques Applied to Semantic Segmentation. *arXiv* **2017**, arXiv:1704.06857.



© 2020 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/>).