*electronics*

MDPI

*Article*

# Real-Time Object Detection in Remote Sensing Images Based on Visual Perception and Memory Reasoning

**Xia Hua [1], Xinqing Wang [1,*], Ting Rui [1], Dong Wang [1,2] and Faming Shao [1]**

[1] College of Field Engineering, PLA Army Engineering University, Nanjing 210007, China; huaxia120888@163.com (X.H.); rtinguu@sohu.com (T.R.); dyhkxydfbb@163.com (D.W.); shaofaming@163.com (F.S.)

[2] Second Institute of Engineering Research and Design, Southern Theatre Command, Kunming 650222, China

* Correspondence: wwwxxxqqq@126.com; Tel.: +86-130-5756-0899

Received: 26 September 2019; Accepted: 8 October 2019; Published: 11 October 2019


**Abstract:** Aiming at the real-time detection of multiple objects and micro-objects in large-scene remote sensing images, a cascaded convolutional neural network real-time object-detection framework for remote sensing images is proposed, which integrates visual perception and convolutional memory network reasoning. The detection framework is composed of two fully convolutional networks, namely, the strengthened object self-attention pre-screening fully convolutional network (SOSA-FCN) and the object accurate detection fully convolutional network (AD-FCN). SOSA-FCN introduces a self-attention module to extract attention feature maps and constructs a depth feature pyramid to optimize the attention feature maps by combining convolutional long-term and short-term memory networks. It guides the acquisition of potential sub-regions of the object in the scene, reduces the computational complexity, and enhances the network's ability to extract multi-scale object features. It adapts to the complex background and small object characteristics of a large-scene remote sensing image. In AD-FCN, the object mask and object orientation estimation layer are designed to achieve fine positioning of candidate frames. The performance of the proposed algorithm is compared with that of other advanced methods on NWPU_VHR-10, DOTA, UCAS-AOD, and other open datasets. The experimental results show that the proposed algorithm significantly improves the efficiency of object detection while ensuring detection accuracy and has high adaptability. It has extensive engineering application prospects.

**Keywords:** remote sensing images; deep learning; neural network; visual perception; object detection

## 1. Introduction

The automatic object detection technology based on remote sensing images is an intelligent data analysis method to realize automatic classification and location of remote sensing objects. It is one of the important research directions in the field of remote sensing image interpretation and has received extensive attention in the civil and military fields. Automatic detection of remote sensing objects is playing an important role in many practical applications such as urban planning, traffic safety, environmental monitoring, and so on. In remote sensing images, besides the serious interference caused by objective factors such as illumination, occlusion, and geometric deformation, multi-class, multi-scale, and multi-directional object detection in complex scenes has always been a crucial and challenging issue in the research of remote sensing image interpretation.

Considering the characteristics of remote sensing images, a new real-time object detection algorithm based on visual perception and convolutional memory network reasoning is proposed for

multi-object and micro-object detection in large-scene remote sensing images. The main work of this paper includes the following two points:

(1) A self-attention pre-screening fully convolutional network (SOSA-FCN) is designed to enhance object self-attention. A self-attention mechanism module is introduced to extract the attention feature map. A depth feature pyramid is constructed on the basis of convolutional long-term and short-term memory networks to reasonably optimize the attention feature map and guide the acquisition of potential object sub-regions in the scene. The low computational complexity enhances the network's ability to extract features of multi-scale objects and adapt to the background complexity and small objects in large scenes of remote sensing images;

(2) An accurate detection fully convolutional network (AD-FCN) is an improved U-Net network. By adding the object mask and the object orientation estimation layer to the traditional U-Net structure, multi-task learning can be achieved, and fine object orientation can be obtained.

## 2. Related Work

Object detection in remote sensing images is a branch of traditional object detection, which has many applications. Common methods of object detection in remote sensing images include template matching, background modeling, and shallow learning. At present, the deep-learning technology has greatly improved the performance of object detection because of its powerful feature representation and end-to-end learning ability. The main methods include R-CNN (regional recommendation convolutional neural network) [1], Faster R-CNN (real-time regional recommendation convolutional neural network) [2], YOLO (Unified Real-Time Object Detection) [3], SSD (single-network multi-scale detector) [4].

Reference [5] proposes a unified self-enhancement network, which is based on the convolutional neural network ($R^2$-CNN) of a remote sensing region. It consists of Tiny-Net, a core network, a middle global attention module, a final classifier, and a detector. Tiny-Net is a lightweight residual network, which can extract depth features quickly and strongly from the input. A global attention module is built in Tiny-Net to suppress false positives. Then, the classifier is used to predict whether there are objects in each candidate box, and if there are objects, the tracking detector is used to locate them accurately. Classifier and detector reinforce each other through end-to-end training, which further speeds up the training process and avoids false alarms.

In reference [6], a cascade convolutional neural network detection framework was designed. The detection framework is composed of two fully convolutional networks: object pre-screening fully convolutional network (P-FCN) and object accurate detection fully convolutional network (D-FCN). P-FCN is a lightweight image classification network, which is responsible for the rapid pre-screening of possible ship areas in large-scene remote sensing images. It has fewer layers, simple training, and less redundancy of candidate frames, which can reduce the computational burden of subsequent networks. D-FCN adds an object mask and a ship orientation estimation layer to the traditional U-Net structure in order to carry out multi-task learning and determine the fine positioning of any ship object.

Reference [7] presents a visual perception object detection algorithm for high-resolution remote sensing images. Firstly, the algorithm obtains the sub-regions of the scene by selective guidance of salient regions and transfers computing resources to the regions that may contain objects, so as to reduce the computational complexity; secondly, it obtains the pre-selected objects by using the object detection model based on the single-detector (YOLO) convolutional neural network; lastly, it proposes an item. Object semantic association suppression is effective in screening pre-selected objects, which can reduce the interference of false objects and the false alarm rate.

Reference [8] proposes a multi-scale convolutional neural network remote sensing object detection framework (MSCNN). This method introduces an enhanced feature pyramid network (EFPN) to enhance the network's ability to extract multi-scale object features. Then, focus classification loss is introduced as a classification loss function to enhance the network's ability to learn difficult samples.

Reference [9] analyzes the influence of pooling operation and object size on regional proposals and proposes a method of combining multi-level features to carry out regional proposals, which improves the recall rate of proposals in object areas. The generation strategy of foreground samples is optimized to avoid invalid foreground samples in the training process, which makes the training of the whole detection model more efficient.

In order to reduce the computational complexity and improve the detection accuracy of large-scale remote sensing images, the method of in-depth learning firstly divides the image into slices by sliding windows and then sends the slices to a neural network model, such as Faster-RCNN and YOLO, for detection. However, there is a lot of redundant information when using the sliding window method, which seriously affects the detection efficiency and greatly depends on the parameters of the sliding window. The phenomena of object omission and object truncation may occur [10,11].

## 3. Theoretical Method

Considering the characteristics of remote sensing images, a new real-time object detection algorithm based on visual perception and convolutional memory network reasoning is proposed for multi-object and micro-object detection in large-scene remote sensing images. As shown in Figure 1, the detection framework is cascaded by two fully convolutional networks, SOSA-FCN and AD-FCN. The overall structure of this model is shown in Figure 1.

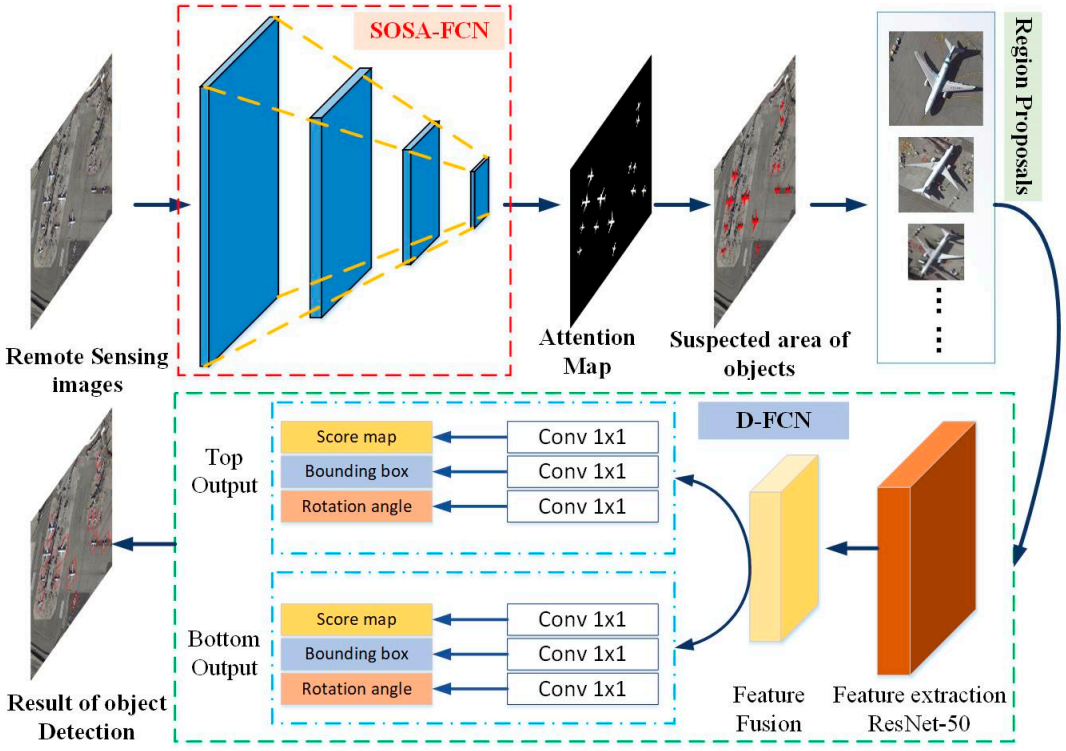


**Figure 1.** Architecture of our Network. The self-attention pre-screening fully convolutional network (SOSA-FCN) realizes pre-screening as in traditional detection methods by using the method of deep learning. After inputting a large-sized remote sensing image into SOSA-FCN, an attention map containing object location information is obtained. According to the attention map, candidate regions suspected of having objects are obtained. The candidate regions are sent to the accurate detection fully convolutional network (AD-FCN) network for accurate object detection. The task of real-time object detection in high-resolution large-scene remote sensing images is realized by using two cascaded networks.

### *3.1. Strengthening Object Self-Attention Pre-Screening Fully Convolutional Network*

Knowledge of the mechanism of attention originates from the study of human vision. In cognitive science, because of the bottleneck of information processing, human beings will selectively pay attention only to part of all available information, ignoring other visible information. The above mechanism is commonly referred to as the attention mechanism. Different parts of the human retina have different degrees of information-processing ability, known as acuteness. The central fovea of the retina has the strongest acuteness. In order to make rational use of limited visual information processing resources, human beings need to select a specific part of the visual area and then focus on it. The attention mechanism has two main aspects: deciding which part of input needs attention and allocating limited information processing resources to important parts [12–14].

The neural attention mechanism allows the neural network to focus on a subset of its inputs (or features), i.e., to select a specific input. Attention can be applied to any type of input regardless of its shape. In the case of limited computing power, the attention mechanism is a resource allocation scheme that is the main means to solve the problem of information overload, allocating computing resources to the most important tasks. In the field of computer vision, the attention mechanism is introduced to process visual information. Attention is a mechanism, or methodology, and there is no strict mathematical definition. For example, traditional local image feature extraction, saliency detection, and sliding window method can all be regarded as attention mechanisms.

In the neural network, the attention module is usually an additional neural network, which can rigidly select some parts of the input or assign different weights to different parts of the input. The attention mechanism in this paper mainly refers to the attention mechanism in neural networks. The self-attention mechanism shows a better balance among the ability to simulate remote dependency, computational efficiency, and statistical efficiency [15]. The self-attention module takes the weighted sum of the features at all positions as the response of the positions, wherein the weight or attention vector is calculated, with a low calculation cost. On the basis of the self-attention model, we have designed a new strengthening object self-attention network (SOSA-Net). The overall structure of the model is shown in Figure 2.

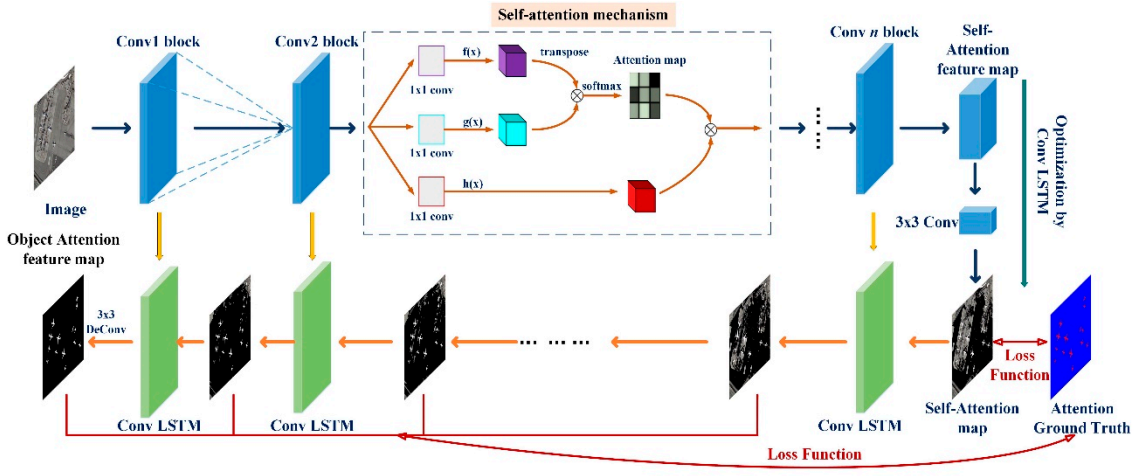


**Figure 2.** Architecture of our SOSA-Net which integrates the visual saliency detection strategy and introduces the conv-LSTM (long-term and short-term memory) network to optimize the self-attention feature graph, making the model's attention to the objects in the image more in line with human eyes' vision.

The self-attention feature map gives a rough but informative prior about visually significant regions. Many previous researches on pixel-labeling tasks have shown that neural networks can combine high-level feature information by encoding at the upper network layer to produce fine labeling results [16–18]. Therefore, we believe that our model can infer more detailed object attention

information based on the self-attention feature map derived from high-order network layer reasoning. After training, the network model aggregates the information of the high-level self-attention feature map and the rich spatial and detailed features of the underlying network through the feature pyramid [19] strategy to focus on and continuously refine the objects in complex backgrounds. As shown in Figure 2, the model is calculated from top to bottom and integrates information from earlier layers in turn. A plurality of conv-LSTM networks (green blocks in the figure) are stacked to construct more meaningful feature expression results by circular connection. We use the sequence and memory characteristics of LSTM to process the features in an iterative manner. At a certain level, conv-LSTM abandons the feature of small information amount and strengthens the feature of large information amount, thus generating a gradually improved inference object reinforcement self-attention feature map [17].

$f(x)$, $g(x)$ and $h(x)$ are common $1 \times 1$ convolutions, the difference is only that the output channel size is different. The output of $f(x)$ is transposed, multiplied by the output of $g(x)$, and normalized by the SoftMax function to obtain an attention map; SAGAN multiplies the obtained attention map and the convolution result obtained through $h(x)$ by pixel [14] to obtain an adaptive attention feature maps. The image features of the previous hidden layer $x \in \mathbb{R}^{C \times N}$ are first converted into two feature spaces $f$, $g$, to calculate attention, where $f(x) = W_f x$, $g(x) = W_g x$.

$$\beta_{j,i} = \frac{\exp\left(s_{ij}\right)}{\sum_{i=1}^{N} \exp\left(s_{ij}\right)} \tag{1a}$$

$$s_{ij} = f(x_i)^T g\left(x_j\right) \tag{1b}$$

where $\beta_{j,i}$ indicates the degree to which the model pays attention to the $i$ position when synthesizing the $j$ region. Then, the output of the attention layer is $= \left(_{1,2}, \ldots, _j, \ldots _N\right) \in \mathbb{R}^{C \times N}$, of which

$$_j = \sum_{i=1}^{N} \beta_{j,i} h(x_i) \tag{2a}$$

$$h(x_i) = W_h x_i \tag{2b}$$

Among the above formulas, $W_f \in \mathbb{R}^{C \times N}$, $W_g \in \mathbb{R}^{C \times N}$, and $W_h \in \mathbb{R}^{C \times N}$ are all weight matrices obtained through learning. In addition, we further multiply the output of the layer of interest by the scale parameter and add it back to the input feature map. Therefore, the final output is:

$$y_i = \gamma_i + x_i \tag{3}$$

in which $\gamma$ is initialized to 0 and then gradually assigns more weight to non-local attention graphs. The network begins to learn some simple goals and then gradually increases the complexity of the task. Conv-LSTM extends the traditional fully connected LSTM to handle spatial features. Basically, this is achieved by using convolution instead of dot product in the LSTM equation. Conv-LSTM has a convolution structure in the input to the state and the state-to-state transition, which can preserve the spatial information of the convolution feature map, thus enabling our network to generate pixel-level labels. Similar to the traditional gate LSTM, conv-LSTM uses memory cells and gates to control the information flow. It works by sequentially updating the internal state $H$ and memory cell $C$ through the values $i$, $f'$, $c$ of three sigmoid gates. In the $t$ step, when the input $X_t$ arrives, if the input gate $i_t$ is activated, the included information of $X_t$ will be accumulated in the memory cell, and if the forgotten gate $f_t'$ is turned on, the state $C_{t-1}$ of the previous memory cell will be forgotten. Whether the latest

cell state $C_t$ should propagate to the final state $H_t$ is further controlled by the output gate $o_t$. Formally, the above memory update process of step $t$ is driven by the following equations:

$$i_t = \sigma\Big(W_i{}^X * X_t + W_i{}^H * H_{t-1} + b_i\Big) \tag{4}$$

$$f_t{}' = \sigma\Big(W_{f'}{}^X * X_t + W_{f'}{}^H * H_{t-1} + b_{f'}\Big) \tag{5}$$

$$o_t = \sigma\Big(W_o{}^X * X_t + W_o{}^H * H_{t-1} + b_o\Big) \tag{6}$$

$$C_t = f'_t \circ C_{t-1} + i_t \circ \tanh\Big(W_c{}^X * X_t + W_c{}^H * H_{t-1} + b_c\Big) \tag{7}$$

$$H_t = o_t \circ \tanh(C_t) \tag{8}$$

where $*$ represents a convolution operation, $\circ$ represents an element-wise product of related elements, $\sigma$ and tanh are activation functions of logical sigmoid and hyperbolic tangent. Input $X_t$, memory cell $C_t$, hidden states $H_t$, and gates $i_t$, $f_t{}'$, $c_t$ are 3D tensors with the same spatial dimension. $W$ and $b$ are learned weights and biases. In this model, conv-LSTM takes the feature $X$ extracted from the convolutional neural network (from the last convolution layer before the aggregation layer) as input and generates accurate object saliency features for final object saliency estimation. Because it operates on still images, the input features in all steps are the same. Here, we use the recursive nature of LSTM to iteratively optimize the salient features of static images, instead of using LSTM to model the time dependence of sequence data.

We combine the features of self-attention prior graph $P_s$ and convolution layer as the input of conv-LSTM. In each time step, conv-LSTM is trained, and salient objects are inferred by using fixed information knowledge, and the features are sequentially optimized according to the updated storage unit and hidden state. Therefore, the features are reorganized to better represent the significance of the object. First, we compress the characteristic response of the convolution layer through the convolution layer of multiple filters to reduce the calculation cost and use sigmoid activation to regularize the characteristic response, so that it is within the same range of $P_s$. Then, the self-attention prior graph $P_s$ is connected with the compressed features and input to conv-LSTM. We apply the $1 \times 1$ and $3 \times 3$ combined convolution kernels to the final conv-LSTM output $H$ to obtain the inference object enhanced self-attention feature map $Q$.

In order to evaluate the significance model, several different measurement standards have been proposed. We adopt the real prominent object annotation $S$ proposed in reference [17] and thus we can obtain the conv-LSTM total loss function defined as:

$$\begin{aligned} L_{Sal}(S,Q) = \quad & L_C(S,Q) + \alpha_1 L_P(S,Q) + \alpha_2 L_R(S,Q) + \alpha_3 L_F(S,Q) \\ & + \alpha_4 L_{MAE}(S,Q) \end{aligned} \tag{9}$$

where the balance parameters are set to $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.1$, and $L_C$ is the weighted cross entropy loss function, which is the main loss function of the conv-LSTM model:

$$L_C(S,Q) = \frac{1}{N}\sum\nolimits_x (\vartheta \cdot (1 - s_x) \cdot \log(1 - q_x) + (1 - \vartheta) \cdot s_x \cdot \log q_x) \tag{10}$$

where $N$ represents the total number of image pixels, $s_k \in S$, $q_k \in Q$. $\vartheta$ refers to the ratio of $S$ significant pixels in the real value, and the weighted cross entropy loss function handles the imbalance between prominent and non-prominent pixels. $L_P$, $L_R$, $L_F$ are used to calculate the similarity of precision, recall, and F-measure scores:

$$L_P(S,Q) = -\frac{\sum_x s_x \cdot q_x}{\sum_x q_x + \varepsilon} \tag{11}$$

$$L_R(S,Q) = -\frac{\sum_x s_x \cdot q_x}{\sum_x s_x + \varepsilon} \tag{12}$$

$$L_F(S,Q) = -\frac{\left(1+\beta^2\right)\cdot L_P(S,Q)\cdot L_R(S,Q)}{\beta^2\cdot L_P(S,Q) + L_R(S,Q) + \varepsilon} \tag{13}$$

where $\beta^2 = 0.3$ is the setting according to reference [17], and $\varepsilon$ is a regularization constant. Because precision, recall, and F-measure are similarity measures, higher values are better, so negative values are used to minimize precision, recall, and F-measure. $L_{MAE}$ is derived from the mean absolute error (MAE) metric, which calculates the difference between the significance graph $Q$ and the truth graph $S$.

$$L_{MAE}(S,Q) = \frac{1}{N}\sum_x \left|s_x - q_x\right| \tag{14}$$

After obtaining the object saliency map $Q$ inferred from the attention prior map $P$, we unsample $Q$ and feed it to the next conv-LSTM to obtain the compression feature from the conv $n-1$ layer for a more detailed optimization. The above process iterates layer by layer to the conv 1 layer. In short, the model can effectively infer the salient features of the learning object, which is due to (1) the learnable self-attention mechanism, (2) iteratively updating the salient features and the cyclic architecture, and (3) effectively merging the spatial rich information from the lower layers in a top-down manner.

### 3.2. *Accurate Object Detection Fully Convolutional Network (AD-FCN)*

AD-FCN is an improved U-Net [20] structure, and Figure 3 shows the structure diagram of AD-FCN.

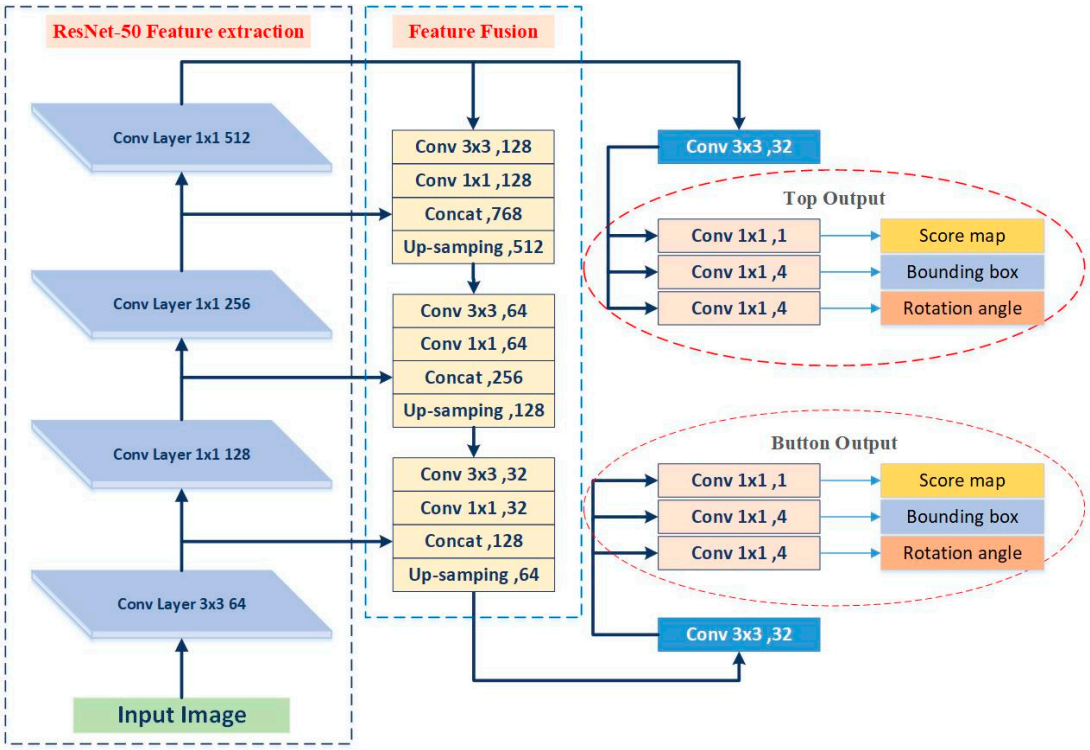


**Figure 3.** Architecture of our AD-FCN. AD-FCN is mainly divided into three parts: feature extraction, feature fusion, and result output. In order to avoid the problems of over-fitting and gradient explosion that may occur in the training process, the initialization of the feature extraction network was completed by using the transfer learning method and the ResNet50 [21] model for reference. In the feature fusion part, the U-Net idea was used for reference. On the basis of FCN, the high-dimensional and low-dimensional features in the convolutional network are fused to realize pixel-level classification of images and improve detection accuracy. In the output part, the confidence score map of one channel, the rectangular frame boundary information map of four channels, and the object rotation angle map of one channel are obtained through three $1 \times 1$ convolution layers [6].

In the traditional U-Net model, the underlying features in the network are compared with the original image to construct a loss function, and then the model parameters are iteratively updated by using a back-propagation algorithm. However, according to the principle of the back propagation algorithm, the parameters closest to the loss constraint in the model will be updated preferentially, and the update amplitude of other parameters will gradually decrease with the propagation distance lengthening, which leads to the traditional U-Net model emphasizing the update of the bottom parameters while training, so that the optimization degree of the top parameters is relatively poor. According to this, this paper introduces additional loss constraints at the top of the model, so that parameters at different levels can be better optimized, thus further improving the detection accuracy.

Two parameters are needed for the location of the rotating rectangular frame: the orientation angle of the object and the boundary information of the rectangular frame. In the AD-FCN network, firstly, the object in the training data is labeled clockwise with a rotating rectangular frame, and then the object mask is generated according to the labeling information, as shown in Figure 4.

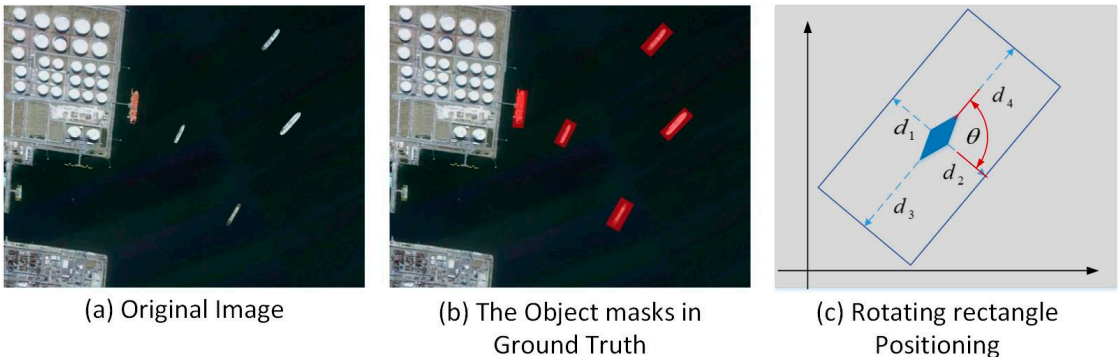


**Figure 4.** Location of object mask and rotating rectangular frame. (**a**) Original image, (**b**) Object mask generated according to the true coordinates, (**c**) positioning sketch of the rotating rectangular frame.

Each pixel in the object mask is defined as $G = \{d_1, d_2, d_3, d_4, \theta\}$. The rotation angle and boundary information of the rectangular box can be obtained. Among them, $d_i$ represents the distance from the pixels to the four sides of the rotating rectangular frame, $\theta$ is the rotating angle of the rectangular frame, that is, the orientation of the object. The angle between the head and the tail lines of the object and the horizontal direction of the image are defined.

In order to realize object detection in an arbitrary rotation direction, AD-FCN adds top loss besides original bottom loss. Therefore, the total loss function in AD-FCN is as follows:

$$L_{\text{D-FCN}} = L_{\text{Top}} + L_{\text{Bottom}} \tag{15}$$

where $L_{\text{Bottom}}$ is the loss value of the bottom output result and the truth map, $L_{\text{Top}}$ is the loss value of the top output result and the corresponding low-resolution truth map. The calculation process of top loss is the same as that of bottom loss. Taking bottom constraint as an example, its loss constraint can be divided into classified loss and geometric loss.

$$L_{\text{Bottom}} = L_{\text{cls}} + L_{\text{geo}} \tag{16}$$

where $L_{\text{cls}}$ represents the loss of classification. In U-Net prediction map, the value of each pixel represents the confidence score of the ship. In view of the limitation of the fixed confidence threshold, which is not flexible enough, the fuzzy adaptive threshold method is used to adjust the adaptive threshold strategy to reduce the false alarm rate and the missed detection rate [22].

By default, $N$ pixels are sent to U-Net by detecting an image. Finally, $M$ pixels are obtained for each pixel to represent the confidence of $M$ categories. Therefore, a total of $N$, $M \times 1$ arrays can be obtained. Take out the maximum values in each array, sort them from big to small, and discard

the values less than 0.1 (if all the values of $N$ are less than 0.1, then there is no object), get the array $C$ of $N \times 1$; $\mu(x)$ is the membership function, $\mu(C_k)$ is the membership degree of the region whose confidence is $C_k$ in array $C$. The ambiguity rate $\gamma(C)$ of array $C$ is a measure of the ambiguity of the array $C$. If $h(C_k)$ is the number of elements of confidence degree $C_k$ in the array $C$, the ambiguity rate $\gamma(C)$ of the array $C$ is defined as follows:

$$\gamma(C) = \frac{2}{n}\sum_{k=0}^{n-1} T(C_k)h(C_k) \tag{17}$$

where $T(C_k) = \min\{\mu(C_k), 1-\mu(C_k)\}$. The ambiguity rate $\gamma(C)$ of the array $C$ depends on the membership function $\mu(x)$, if the membership function is the $S$ function, that is

$$\mu(x) = \begin{cases} 0, & 0 \le x \le q - \Delta q \\ 2\left[\frac{(x-q+\Delta q)}{2\Delta q}\right]^2, & q - \Delta q \le x \le q \\ 1 - 2\left[\frac{(x-q+\Delta q)}{2\Delta q}\right]^2, & q < x \le q + \Delta q \\ 1, & q + \Delta q < x \le C_n \end{cases} \tag{18}$$

where $L_{\text{cls}}$ represents the loss of classification. In the U-Net prediction map, the value of each pixel represents the confidence score of the ship.

Then $\mu(x)$ is determined by the window width $c = 2\Delta q$ and the parameter $q$. Once the window width is selected, $\gamma(C)$ is only related to the parameter $q$. The solution process of the fuzzy threshold method is to set the window width in advance, while the coefficient is always set at 0.3. By changing $q$, the membership function $\mu(x)$ slides on the confidence interval $[C_0, C_{n-1}]$, and the fuzzy rate curve is obtained by calculating the fuzzy rate $\gamma_q(C)$. The valley point of the curve, even if $\gamma_q(C)$ has the minimum value $q$, is the adaptive threshold required. In this paper, the value of confidence greater than $q$ is set to 255, and the predictive mask is obtained. Using Diss's loss method, a classification loss function is constructed by comparing the predictive mask with the real mask, in which $y_{\text{cls}}$ represents the real mask, and $\hat{y}_{\text{cls}}$ represents the predictive mask.

$$L_{\text{cls}} = \frac{2\left|y_{\text{cls}} \cap \hat{y}_{\text{cls}}\right|}{\left|y_{\text{cls}}\right| + \left|\hat{y}_{\text{cls}}\right|} \tag{19}$$

The geometric loss $L_{\text{geo}}$ includes rectangular frame positioning loss and rotation angle loss, as shown in the formula

$$L_{\text{geo}} = L_{\text{Bbox}} + L_{\theta} \tag{20}$$

where, $L_{\text{Bbox}}$ is the loss of the rectangular frame. IoU is used to calculate the loss of the rectangular frame. Its expression is shown in Formula (21), where $\hat{R}$ represents the predicted mask area, and $R$ represents the real mask area.

$$L_{\text{Bbox}} = -\log \text{IoU}\left(\hat{R}, R\right) = -\log \frac{\left|\hat{R} \cap R\right|}{\left|\hat{R} \cup R\right|} \tag{21}$$

$L_{\theta}$ is the loss of rotation angle, and the calculation method is based on the formula:

$$L_{\theta} = 1 - \cos\left(\hat{\theta} - \theta\right) \tag{22}$$

where, $\hat{\theta}$ represents the predicted rotation angle, and $\theta$ represents the real rotation angle.

## 4. Results

In this chapter, we implement ablation research to analyze and verify the impact of different computing components on detection strategies. At the same time, in order to verify the overall advancement of this model, we compare the most advanced object detection models based on deep learning that have been proposed in recent years.

### *4.1. Experimental Dataset and Evaluation Index*

The image data used in this experiment are from a high-resolution optical remote sensing image. The classifier based on a conventional dataset training has poor detection ability with high-resolution remote sensing images. The main reason is that a remote sensing image has particular features, such as various scales, a special perspective, small objects, multi-directionality, and high background complexity. In this paper, several mainstream remote sensing image databases are used as experimental datasets, including DOTA [23], UCAS-AOD [24], NWPU VHR-10 [25], TGRS-HRRSD-Dataset [26], and RSOD [27]. In addition, stretched image histogram [coefficient 0.02], rotation, and random flip are used to expand the dataset.

Considering a small object in large-scale remote sensing image and the need of practical engineering application, object detection needs to perform the two tasks of object location and object recognition simultaneously. The accuracy of object location is determined by comparing the overlap degrees (IoU) of the predicted border and ground truth border and the size of the threshold, and the correctness of object recognition is determined by comparing the confidence score and the threshold. The above two steps are used to determine whether the object detection is correct or not. Finally, the problem of multi-class object detection is transformed into a two-class problem of correct detection of a certain kind of object and detection error. Thus, a confusion matrix can be constructed, and a series of indicators of object classification can be used to evaluate the accuracy of the model [28].

In multi-object classification, the recognition effect of existing objects is concerned, and the recognition rate generally refers to the detection rate. The sum of sham alarm rate $P_f$, detection rate $P_d$, missed detection rate $P_m$, and false detection rate $P_e$ is 1. In the actual calculation, the recognition rate is first calculated, then the false alarm rate and the false alarm rate are calculated, and the classified false alarm rate is calculated by counting the object types recognized by the remaining systems but not actually present. The false alarm rate accumulated in a certain period of time should be calculated for the false alarm rate in multi-object recognition. For datasets, we use the averaging method to calculate the overall sham alarm rate, missed detection rate, detection rate, and false detection rate [28].

Deep learning adjusts the weight of a neural network through the back propagation of errors to achieve the goal of modeling. The number of back propagation iterations gradually increased from tens of thousands to hundreds of thousands until the training error tended to converge. Finally, the model is evaluated by calculating the average precision (AP) of the model on the test set and the mean AP (m AP) of all categories. AP measures the accuracy of the detection algorithm from the perspective of recall rate and accuracy rate. AP is the most intuitive standard to evaluate the accuracy of depth detection model, which can be used to analyze the detection effect of a single category. MAP is the average value of each category of AP. The higher the mAP, the higher the comprehensive detection performance of the model in all categories [29].

Because of different picture sizes, the testing speed of individual pictures varies greatly, so the unit of testing time is set to $s/1000 \times 1000$, that is, the average testing time per $1000 \times 1000$ size images. In this paper, the test platform CPU is Intel Core i7 6700, the deep learning operating environment is TensorFlow 2.0, cuDNN 7.4.1, CUDA 10.0, Python version 3.7, the graphics card is NVIDIA Titan X, and the operating system is Ubuntu x64. In this paper, the initial learning rate of the network is set to 0.003, and when the number of training iterations reaches 100 and 20,000, the rate decreases 10 times. The activation function used is Leaky. We randomly initialize all layers by extracting weights from a zero-mean Gaussian distribution with a standard deviation of 0.01.

### *4.2. Analysis of the Pre-screening Effect of SOSA-FCN*

To evaluate the effectiveness of SOSA-FCN in visual attention, we first verify the performance of SOSA-FCN for FP (fixation prediction) tasks. The purpose of this experiment is to study the validity of the fixed map in advance learning, rather than compare it with the most advanced FP model. Then, we evaluate the performance of SOSA-FCN in primary salient object detection (SOD) tasks. For FP tasks, we use five typical measures: normalized scanpath saliency (NSS), similarity metric (SIM), linear correlation coefficient (CC), AUC-Judd, and shuffled AUC. For SOD tasks, three standard metrics are used, namely, precision and recall (P-R) curve, F-measure, and MAE [17]. The image zooming process is added before the SOSA-FCN network test. By setting the resize parameter value, the detection speed can be significantly improved with a certain detection accuracy. We set the resize parameter to 3.

We evaluate the fixed prior map generated by SOSA-FCN and compare it with several state-of-the-art fixed models, including three classical models, i.e., ITTI [30], GBVS [31], and AIM [32], and the in-depth learning-based models P-FCN [6], Mr-CNN [33], Shallow-Net [34], Deep-Net [34], and AS-Net [17], reporting the results through DOTA and NWPU VHR-10 datasets.

As shown in Table 1, SOSA-FCN performs better than previous non-in-depth learning models and is more competitive in attention detection and efficiency than the current best-performing in-depth learning competitors, which indicates that the proposed SOSA-FCN may obtain better FP results with more detailed spatial information.

**Table 1.** Quantitative comparison of different fixation prediction (FP) models using different datasets.

| **Methods** | **AUC-Judd** | | **SIM** | | **Shuffled AUC** | |
|---|---|---|---|---|---|---|
| | **DOTA** | **NWPU VHR-10** | **DOTA** | **NWPU VHR-10** | **DOTA** | **NWPU VHR-10** |
| ITTI | 0.48 | 0.61 | 0.18 | 0.17 | 0.43 | 0.55 |
| GBVS | 0.57 | 0.71 | 0.22 | 0.23 | 0.45 | 0.58 |
| AIM | 0.52 | 0.64 | 0.2 | 0.19 | 0.48 | 0.66 |
| Mr-CNN | 0.63 | 0.76 | 0.23 | 0.32 | 0.56 | 0.71 |
| P-FCN | 0.75 | 0.83 | 0.31 | 0.38 | 0.61 | 0.75 |
| Shallow-Net | 0.59 | 0.73 | 0.25 | 0.29 | 0.58 | 0.69 |
| Deep-Net | 0.74 | 0.81 | 0.28 | 0.33 | 0.55 | 0.71 |
| AS-Net | 0.76 | 0.85 | 0.36 | 0.41 | 0.62 | 0.74 |
| SOSA-FCN | 0.78 | 0.87 | 0.41 | 0.46 | 0.66 | 0.75 |
| | CC | | NSS | | Speed(s/1000x1000) | |
| | DOTA | NWPU VHR-10 | DOTA | NWPU VHR-10 | DOTA | NWPU VHR-10 |
| | 0.23 | 0.31 | 0.82 | 1.05 | 4.343 | 3.969 |
| | 0.32 | 0.39 | 1.13 | 1.32 | 3.532 | 4.032 |
| | 0.16 | 0.22 | 0.71 | 0.98 | 5.318 | 4.893 |
| | 0.28 | 0.37 | 1.09 | 1.29 | 1.028 | 0.798 |
| | 0.36 | 0.46 | 1.18 | 1.33 | 0.258 | 0.336 |
| | 0.31 | 0.35 | 1.27 | 1.47 | 1.325 | 1.135 |
| | 0.37 | 0.41 | 1.22 | 1.41 | 0.983 | 0.632 |
| | 0.35 | 0.45 | 1.31 | 1.52 | 0.452 | 0.398 |
| | 0.42 | 0.51 | 1.29 | 1.49 | 0.175 | 0.149 |

Here, we evaluate the performance of SOSA-FCN on its main task: SOD. Three widely used datasets, DOTA, UCAS-AOD, and NWPU VHR-10, were quantitatively studied. We compare SOSA-Net with alternative methods based on in-depth learning: P-FCN, ELD [35], RFCN [36], DHS [37], HEDS [38], DSSOD [38], AS-Net. We also consider several classical non-in-depth learning models: HS [39], DRFI [40], and DSR [41], using results from authors or running their open source implementations through the original settings. We report in Table 2 the maximum F measurements ($F_\beta$) and MAE scores. Overall, the proposed method achieves better performance on three datasets using all evaluation indicators.

**Table 2.** The $F_\beta$ and mean absolute error (MAE) scores of salient object detection (SOD) for different datasets.

| Methods | DOTA | | UCAS-AOD | | NWPU VHR-10 | |
|---|---|---|---|---|---|---|
| | $F_\beta$ | MAE | $F_\beta$ | MAE | $F_\beta$ | MAE |
| DRFI | 0.612 | 0.358 | 0.703 | 0.298 | 0.731 | 0.255 |
| HS | 0.572 | 0.459 | 0.611 | 0.376 | 0.769 | 0.223 |
| P-FCN | 0.765 | 0.153 | 0.804 | 0.158 | 0.842 | 0.138 |
| RFCN | 0.773 | 0.127 | 0.799 | 0.101 | 0.806 | 0.092 |
| DHS | 0.798 | 0.098 | 0.821 | 0.089 | 0.851 | 0.073 |
| ELD | 0.735 | 0.112 | 0.783 | 0.113 | 0.803 | 0.096 |
| HEDS | 0.803 | 0.089 | 0.832 | 0.091 | 0.858 | 0.085 |
| AS-Net | 0.815 | 0.065 | 0.845 | 0.078 | 0.868 | 0.063 |
| DSSOD | 0.814 | 0.086 | 0.851 | 0.076 | 0.876 | 0.066 |
| SOSA-FCN | 0.818 | 0.061 | 0.856 | 0.078 | 0.881 | 0.049 |

In this paper, the P-R curve is used to compare the proposed method with the existing saliency detection method based on depth learning. In Figure 5, we depict P-R curves generated by the method used in this paper and the most advanced method used in the past on three popular datasets, i.e., DOTA, UCAS-AOD, and NWPU VHR-10. Obviously, by combining the results of multiple datasets, the algorithm in this paper can achieve the best results. We also find that when the recall score is close to 1, the accuracy of our method is much higher, which reflects the fact that our false alarm is much lower than that of other methods and shows the effectiveness of our strategy. The visual attention map of the object in the high-resolution remote sensing image with complex background thus obtained more closely corresponds to the basic facts.

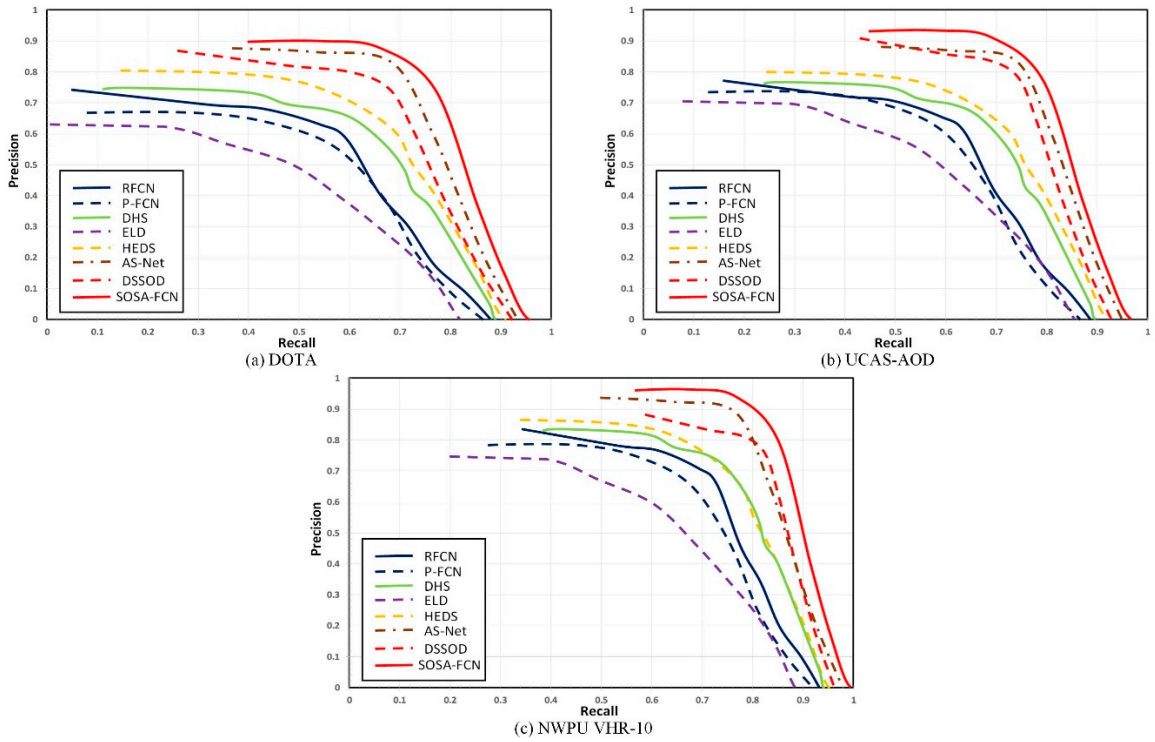


**Figure 5.** SOD results with precision and recall (P-R)-curve using three datasets.

Figure 6 shows the qualitative results for an example image from the above dataset, which verify that the proposed SOSA-FCN is very suitable for the task of extracting attention maps of high-resolution complex scenes.

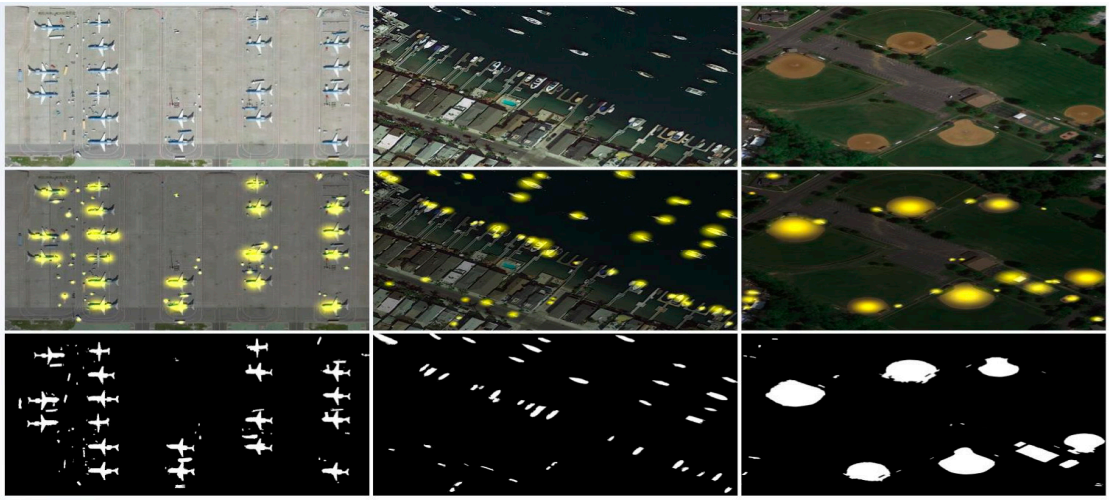

**Figure 6.** Qualitative results of SOSA-FCN. The first row shows the original images, the second row shows the fixation maps of the object, and the third row shows the attention feature maps of the object.

### *4.3. Analysis of the Detection Effect of AD-FCN*

The experiment was carried out on the NWPU VHR-10 dataset. Firstly, we compare the proposed method with the seven mainstream baseline detection strategies of SS-A [42], SS-V [43], SS-A-F [44], EB-A-F [45], EB-V-F [46], EB-V-F-CS [29], and EB-V-F-BR [29]. The results are shown in Table 3. The evaluation index is mAP (mean average precision).

**Table 3.** Comparison among the proposed framework and other baselines detection strategies using the NWPU VHR-10 set in terms of mAP.

| Strategies | Aircraft | Ship | Storage Tank | Ballpark | Tennis Court | Basketball Court |
|---|---|---|---|---|---|---|
| SS-A | 36.31 | 23.12 | 46.28 | 32.35 | 36.72 | 34.71 |
| SS-V | 44.72 | 25.41 | 54.82 | 36.21 | 42.83 | 42.62 |
| SS-A-F | 41.81 | 24.15 | 52.43 | 34.62 | 38.42 | 36.18 |
| EB-A-F | 38.53 | 26.16 | 51.31 | 32.55 | 43.25 | 37.33 |
| EB-V-F | 43.24 | 38.23 | 66.54 | 38.73 | 46.31 | 47.15 |
| EB-V-F-CS | 51.72 | 32.81 | 56.81 | 40.86 | 52.28 | 49.82 |
| EB-V-F-BR | 47.85 | 35.41 | 63.52 | 42.91 | 52.47 | 55.57 |
| AD-FCN | 69.38 | 61.82 | 69.66 | 62.58 | 61.23 | 73.21 |
| | Athletic field | Port | Bridge | Car | Avg. | |
| | 35.76 | 26.22 | 19.34 | 13.61 | 30.442 | |
| | 43.77 | 33.85 | 25.42 | 22.93 | 37.258 | |
| | 41.93 | 29.51 | 27.65 | 19.32 | 34.602 | |
| | 40.12 | 28.42 | 24.81 | 21.84 | 34.432 | |
| | 46.75 | 35.13 | 34.37 | 29.52 | 42.597 | |
| | 38.61 | 38.62 | 31.22 | 23.81 | 41.656 | |
| | 47.47 | 39.85 | 36.83 | 37.26 | 45.914 | |
| | 75.28 | 57.83 | 53.77 | 56.38 | 64.114 | |

By comparing the experimental results of various detection strategies, we find that, compared with other mainstream detection strategies, AD-FCN introduces additional loss constraints at the top layer of the model, so that the parameters at different levels can be better optimized; also, an improved strategy consisting in adding an object mask and an object orientation estimation layer for multi-task learning can effectively improve the detection accuracy.

In order to further analyze the positioning error of AD-FCN, we classify each extraction object in positive training images as one of the following five situations: correct positioning (overlap (>50%), recommendation (complete) in real internal ground truth, recommendation containing real knots.

None of the above situations is ground truth in origin, but no-zero overlap and no overlap [47]. Figure 7 shows the frequency of these five situations for each object class and the average error for all categories of AD-FCN. The experimental results show that the positioning accuracy of AD-FCN for different types of objects can reach more than 70%, which basically avoids the problem of no overlap error. Figure 7 shows that nearly 98% of the selected object detection frames overlap with their corresponding basic fact boundaries to some extent.

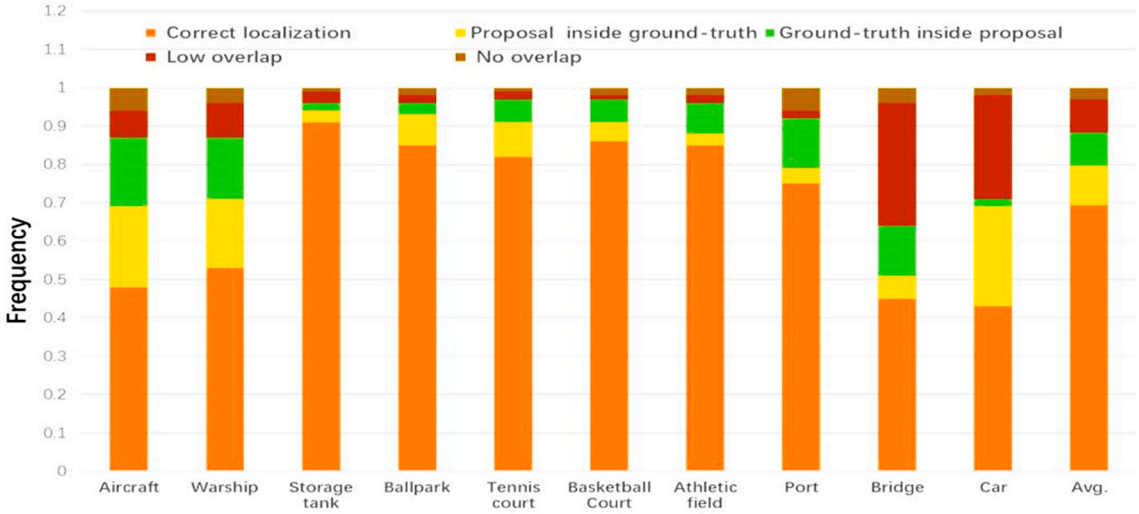


**Figure 7.** Per-class frequency of error modes as well as average error across all classes for the AD-FCN.

### *4.4. Comparison Experiments with State-of-the-Art Models*

In order to verify the effectiveness of our proposed architecture, we use the most advanced object detection models based on in-depth learning tha have been used in recent years, i.e., R2-CNN [5], FPN Faster R-CNN [5], C-SPCL [47], F-RLN [48], MSCNN [8], DSSD [49], SDBD [50], YOLOv3 [51], VPN [7], MDSSD [52] as comparison algorithms, which are higher in DOTA, TGRS-HRD-Data. The results of multi-object detection are compared using the optical remote sensing image dataset of resolution. The above comparison algorithm uses the default parameters set in the official code published by the author.

The experimental results on the DOTA dataset are shown in Table 4. Compared with the results of the algorithm in this paper and those of other deep learning algorithms in the table, mAP for our model is improved by about 5%–38%, and the detection rate is increased by 2%–42% for the DOTA dataset. The detection speed is obviously higher than that of other algorithms, which basically meets the real-time requirements.

**Table 4.** Comparison of the object detection performance using the DOTA dataset.

| Model | Map (%) | MAX RECALL (%) | Pd (%) | Speed (ms) |
|---|---|---|---|---|
| R2-CNN | 87.32 | 97.18 | 83.03 | 17.63 |
| FPN Faster R-CNN | 83.04 | 93.05 | 69.85 | 24.26 |
| MSCNN | 84.65 | 91.86 | 81.02 | 33.48 |
| DSSD | 54.27 | 83.86 | 43.65 | 53.27 |
| SDBD | 79.19 | 92.38 | 79.84 | 35.73 |
| F-RLN | 82.53 | 90.11 | 80.52 | 46.18 |
| C-SPCL | 85.21 | 96.33 | 86.12 | 29.58 |
| YOLOv3 | 65.68 | 87.12 | 57.98 | 47.25 |
| VPN | 83.08 | 94.03 | 82.86 | 21.68 |
| MDSSD | 85.38 | 95.13 | 83.21 | 19.45 |
| OURS | 92.35 | 98.82 | 85.75 | 14.03 |

The experimental results for the TGRS-HRRSD dataset are shown in the Table 5. Comparing the detection results of our algorithm with those of other deep learning algorithms, indicated in the table, for the TGRS-HRRSD dataset, mAP is improved by about 3%–38%, the detection rate is improved by 2%–39%, and the detection speed is significantly higher than that of the other algorithms; our algorithm basically meets the requirements of real-time detection.

**Table 5.** Comparison of the object detection performance using the TGRS-HRRSD dataset.

| Model | mAP (%) | MAX RECALL (%) | Pd (%) | Speed (ms) |
|---|---|---|---|---|
| R2-CNN | 91.35 | 98.48 | 88.93 | 16.13 |
| FPN Faster R-CNN | 87.24 | 96.53 | 73.82 | 22.36 |
| MSCNN | 86.53 | 95.46 | 86.72 | 22.18 |
| DSSD | 56.47 | 89.79 | 53.15 | 49.72 |
| SDBD | 83.79 | 97.28 | 85.34 | 48.63 |
| F-RLN | 85.36 | 94.43 | 83.33 | 43.69 |
| C-SPCL | 91.68 | 99.03 | 91.07 | 18.15 |
| YOLOv3 | 69.48 | 94.82 | 64.78 | 35.45 |
| VPN | 89.28 | 93.23 | 88.79 | 17.63 |
| MDSSD | 90.36 | 98.15 | 89.23 | 16.54 |
| OURS | 94.65 | 99.36 | 90.98 | 10.29 |

Figures 8 and 9 are fusion detection results of multiple objects in DOTA and TGRS-HRRSDt datasets, with four categories of objects including: building (yellow border), vehicle (green border), ship (blue Border), aircraft (red border). By analyzing the experimental results, we find that the framework of this paper can complete a relatively fine and accurate detection of multiple objects in high-resolution complex scenes, has strong robustness against interferences such as illumination, shadow, occlusion, etc., and can also give reasonable detection frames for dense and small objects such as vehicles, aircrafts, ships, etc.

**Figure 8.** Results of our Network for three categories of objects (ship, building, car).

**Figure 9.** Results of our Network for three categories of objects (aircraft, building, car).

## 5. Discussion

Although our algorithm has achieved ideal detection results on both datasets considered, in the process of analyzing the experimental results, we found that some distortion and strong illumination interference caused extremely rare missed detection and false detection errors. As shown in Figure 10, some vehicle objects in the shade were not detected. At the same time, the shadow of some vehicles was mis-detected as the object. However, through the follow-up experiments, we found that these problems could be eliminated by adjusting the confidence parameters.

**Figure 10.** Some error results of our network.

## 6. Conclusions

On the basis of the analysis of the limitations of existing algorithms for object detection in large-scale high-resolution remote sensing images and the characteristics of large-scale high-resolution remote sensing images, this paper proposes a cascade convolutional neural network real-time object detection framework for remote sensing images that combines visual perception and convolutional memory network reasoning. The proposed algorithm uses the visual attention mechanism and the convolutional memory network reasoning method and combines the fast and accurate characteristics of a deep learning algorithm. Ablation experiments and comparative experiments were carried out on several public datasets, and relatively advanced detection results were obtained. It was verified that the cascade network structure proposed in this paper can achieve a faster detection speed compared with other methods on the premise of ensuring detection accuracy and can basically meet the detection requirements of real-time detection and accuracy in engineering practice.

**Author Contributions:** Conceptualization, X.H. and X.W.; methodology, X.H., X.W. and T.R.; software, X.H., D.W. and F.S.; validation, X.H., X.W., and T.R.; formal analysis, X.H. and T.R.; investigation D.W. and F.S.; data curation, X.W.; writing—original draft preparation, X.H., X.W., T.R., D.W., and F.S.; visualization X.H., X.W., and T.R.; project administration, X.H. and X.W.

**Funding:** This work was supported in part by the China National Key Research and Development Program (No. 2016YFC0802904), National Natural Science Foundation of China (61671470), Natural Science Foundation of Jiangsu Province (BK20161470), 62nd batch of funded projects of China Postdoctoral Science Foundation (No. 2017M623423).

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

## References

1. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, 23–28 June 2014; pp. 580–587.
2. Ren, S.; He, K.; Girshick, R.B.; 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]
3. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
4. 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 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
5. Pang, J.; Li, C.; Shi, J.; Xu, Z.; Feng, H. $R^2$-CNN: Fast Tiny Object Detection in Large-scale Remote Sensing Images. *arXiv Comput. Vis. Pattern Recognit.* **2019**, *1902*, 06042.
6. Chen, H.; Liu, Z.; Guo, W.; Zhang, Z.; Yu, W. Fast detection of ship targets in large-scale remote sensing images based on cascade convolution neural network. *J. Radar* **2019**, *8*, 413–424.
7. Li, C.; Zhang, Y.; Lan, T.; Du, Y. A visual perception target detection algorithm for high resolution remote sensing images. *J. Xi'an Jiaotong Univ.* **2018**, *6*, 9–16. (In Chinese)
8. Yao, Q.; Hu, X.; Le, H. Remote sensing target detection based on multi-scale convolution neural network. *J. Opt.* **2019**, 1–11. (In Chinese)
9. Wang, L.; Feng, Y.; Zhang, M. Optical remote sensing image target detection method. *Syst. Eng. Electron. Technol.* **2019**, *41*, 1–8. (In Chinese)
10. Zhang, Z.; Guo, W.; Zhu, S.; Yu, W. Toward Arbitrary-Oriented Ship Detection with Rotated Region Proposal and Discrimination Networks. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 1745–1749. [CrossRef]
11. Zhao, J.; Guo, W.; Zhang, Z.; Yu, W. A coupled convolutional neural network for small and densely clustered ship detection in SAR images. *Sci. China Ser. F Inf. Sci.* **2019**, *62*, 42301. [CrossRef]
12. Orhan, F.; Cho, K.; Bengio, Y. Multi-Way, Multilingual Neural Machine Translation with a Shared Attention Mechanism. *arXiv* **2016**, arXiv:1601.01073.

13. Choi, E.; Bahadori, M.T.; Sun, J.; Kulas, J.; Schuetz, A.; Stewart, W. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In Proceedings of the Advances in Neural Information Processing Systems 29 (NIPS 2016), Barcelona, Spain, 5–10 December 2016.
14. Chu, Q.; Ouyang, W.; Li, H.; Wang, X.; Liu, B.; Yu, N. Online Multi-Object Tracking Using CNN-based Single Object Tracker with Spatial-Temporal Attention Mechanism. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
15. Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. Self-Attention Generative Adversarial Networks. *ArXiv Mach. Learn.* **2018**, *1805*, 08318.
16. Lakatos, P.; Musacchia, G.; Connel, M.; Falchier, A.; Javitt, D.; Schroeder, C. The spectrotemporal filter mechanism of auditory selective attention. *Neuron* **2013**, *77*, 750–761. [CrossRef] [PubMed]
17. Wang, W.; Shen, J.; Dong, X.; Borji, A. Salient Object Detection Driven by Fixation Prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 1711–1720.
18. Salimans, T.; Goodfellow, I.; Zaremba, W. Improved Techniques for Training GANs. *arXiv* **2016**, arXiv:1606.03498.
19. Kodali, N.; Abernethy, J.; Hays, J.; Kira, Z. On Convergence and Stability of GANs. *arXiv* **2018**, arXiv:1705.07215.
20. Ronneberger, O.; Philipp, F.; Thomas, B. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Berlin, Germany, 2015; pp. 234–241.
21. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
22. Hua, X.; Wang, X.; Wang, D.; Ma, Z.; Shao, F. Multi-objective detection of traffic scene based on improved SSD. *J. Opt.* **2018**, 221–231.
23. Xia, G.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A Large-scale Dataset for Object Detection in Aerial Images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017. (In Chinese).
24. Liu, C.; Ke, W.; Qin, F.; Ye, Q. Linear Span Network for Object Skeleton Detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 136–151.
25. Cheng, G.; Zhou, P.; Han, J. Learningrotation-invariant convolutional neural networks for object detection in VHRoptical remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 7405–7415. [CrossRef]
26. Zhang, Y.; Yuan, Y.; Feng, Y.; Lu, X. Hierarchical and Robust Convolutional Neural Network for Very High-Resolution Remote Sensing Object Detection. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 5535–5548. [CrossRef]
27. Long, Y.; Gong, Y.; Xiao, Z.; Liu, Q. Accurate Object Localization in Remote Sensing Images Based on Convolutional Neural Networks. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 2486–2498. [CrossRef]
28. Wang, X.; Hua, X.; Xiao, F.; Li, Y.; Hu, X.; Sun, P. Multi-Object Detection in Traffic Scenes Based on Improved SSD. *Electronics* **2018**, *7*, 302. [CrossRef]
29. Zhang, D.; Han, J.; Zhao, L.; Meng, D. Leveraging Prior-Knowledge for Weakly Supervised Object Detection Under a Collaborative Self-Paced Curriculum Learning Framework. *Int. J. Comput.* **2018**, *127*, 363–380. [CrossRef]
30. Laurent, I.; Koch, C.; Niebur, E. A Model of Saliency-based Visual Attention for Rapid Scene Analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **1998**, *20*, 1254–1259.
31. Zhao, X.; Chen, B.; Pei, L.; Li, T.; Li, M. Hierarchical saliency: A new salient target detection framework. *Int. J. Control Autom. Syst.* **2016**, *14*, 301–311. [CrossRef]
32. Bruce, N.D.; Tsotsos, J.K. Saliency, attention, and visual search: An information theoretic approach. *J. Vis.* **2009**, *9*, 5. [CrossRef] [PubMed]
33. Liu, N.; Han, J.; Liu, T.; Li, X. Learning to Predict Eye Fixations via Multiresolution Convolutional Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 392–404. [CrossRef] [PubMed]
34. Pan, J.; Sayrol, E.; Giro-i-Nieto, X.; McGuinness, K.; O'Connor, N.E. Shallow and Deep Convolutional Networks for Saliency Prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 598–606.

35. Lee, G.; Tai, Y.W.; Kim, J. Deep Saliency with Encoded Low level Distance Map and High Level Features. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
36. Wang, L.; Wang, L.; Lu, H.; Zhang, P.; Ruan, X. Saliency Detection with Recurrent Fully Convolutional Networks. In *European Conference on Computer Vision*; Springer: Berlin, Germany, 2016.
37. Liu, N.; Han, J. DHSNet: Deep Hierarchical Saliency Network for Salient Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
38. Hou, Q.; Cheng, M.M.; Hu, X.; Broji, A.; Tu, Z.; Torr, P.H. Deeply Supervised Salient Object Detection with Short Connections. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
39. Yan, Q.; Xu, L.; Shi, J.; Jia, J. Hierarchical Saliency Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, ON, USA, 23–28 June 2013; pp. 1155–1162.
40. Jiang, H.; Wang, J.; Yuan, Z.; Wu, Y.; Zheng, L.; Li, S. Salient Object Detection: A Discriminative Regional Feature Integration Approach. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2083–2090.
41. Li, X.; Lu, H.; Zhang, L.; Ruan, X.; Yang, M.H. Saliency Detection via Dense and Sparse Reconstruction. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 2976–2983.
42. Uijlings, J.R.R.; Van De Sande, K.E.; Gevers, T.; Smeulders, A.W. Selective Search for Object Recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [CrossRef]
43. Song, G.; Leng, B.; Liu, Y.; Hetang, G.; Cai, S. Region-based Quality Estimation Network for Large-scale Person Re-identification. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2017.
44. Ballester, P.; Araujo, R.M. On the performance of GoogLeNet and AlexNet applied to sketches. In Proceedings of the Thirtieth Aaai Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016.
45. Zitnick, C.L.; Dollár, P. Edge Boxes: Locating Object Proposals from Edges. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014.
46. Shi, G.; Xie, X.; Han, X.; Liao, Q. Visualization and Pruning of SSD with the base network VGG16. In Proceedings of the International Conference on Deep Learning Technologies, Chengdu, China, 2–4 June 2017.
47. Cinbis, R.G.; Jakob, V.; Cordelia, S. Weakly Supervised Object Localization with Multi-Fold Multiple Instance Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 189–203. [CrossRef]
48. Han, Z.; Zhang, H.; Zhang, J.; Hu, X. Fast aircraft detection based on region locating network in large-scale remote sensing images. In Proceedings of the international conference on image processing, Beijing, China, 17–20 September 2017; pp. 2294–2298.
49. Fu, C.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A. DSSD: Deconvolutional Single Shot Detector. arXiv: Computer Vision and Pattern Recognition. *arXiv* **2017**, arXiv:1701.06659.
50. Tong, L.; Zhang, J.; Lu, X.; Zhang, Y. SDBD: A Hierarchical Region-of-Interest Detection Approach in Large-Scale Remote Sensing Image. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 699–703.
51. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, *1804*, 02767.
52. Guo, Z.; Song, P.; Zhang, Y.; Yang, M.; Sun, X.; Sun, H. Aircraft Object Detection in Remote Sensing Images Based on Deep Convolution Neural Network. *J. Electron. Inf. Sci.* **2018**, *40*, 149–155. (In Chinese)