2021 International Conference on Computer Information Science and Artificial Intelligence (CISAI) | 978-1-6654-0692-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/CISAI54367.2021.00020

RFNet: Toward High-Quality Object Detection in Aerial Images

Hui Tang[1,a]
[1]School of Information Engineering
Southwest University of Science and Technology
Mianyang, China
[a]huitang@mails.swust.edu.cn

Jing Wu[1,2,*]
[1]School of Information Engineering
Southwest University of Science and Technology
Mianyang, China
[2]Robot Technology Used for Special Environment Key Laboratory of Sichuan Province
Southwest University of Science and Technology
Mianyang, China
[*] Corresponding author: jingwu@swust.edu.cn

***Abstract*—Due to the object overlap and multi-scale transformation problems in aerial image detection, the detection system has false and missed detections. In this paper, we propose feature Re-Fusion Network, named RFNet. Specifically, the recurses multi-layer feature maps into a pyramid network, cross fusion of upper and lower features to enhancing information flow. Next, we introduced a lightweight module that connects feature semantic information for fusion to improve the feature fusion ability. Finally, we optimized the parameters of attention mechanism block to adapt to aerial images detection. We equipped it to the modern detector Cascade R-CNN on the VisDrone dataset, which improved the mean Average Precision(mAP) by 1.1%, increased middle-size object detection accuracy by 2.1%, and increased the accuracy of the truck class by 2.4%.**

Keywords-Feature fusion; pyramid network; attention mechanism; object detection

I. INTRODUCTION

UAV vision has the advantages of wide scope, strong flexibility, and low cost. These are significant advantages in detecting and tracking moving objects, searching and locating in special environments, and constructing intelligent transportation systems. However, due to the acquisition angle and flight altitude, objects emerge at a variable scale and smaller size. This renders the detection algorithm ineffective in general scenarios. The study of fast and accurate object detection for UAV images is an important topic in the field of UAV vision, and it has therefore warranted research from several directions. In 2018, Tianjin University initiated the UAV aerial image object detection competition, which attracted many research institutions and universities and developed abundant object detection algorithms [1]. With the development of datasets and hardware computing capabilities, most current mainstream object detection methods are based on deep learning. Convolutional neural networks (CNNs) learn image features on the training dataset and predict the optimal bounding box.

Region-based CNN (R-CNN) [2] is an important breakthrough in CNN object detection, but training and testing consume a significant amount of time and storage space. In 2015, Ren *et al.* proposed a two-stage object detection algorithm, Faster R-CNN [3], rather than the selective search [4] algorithm for candidate frame selection, enabling the entire network to be trained end-to-end. In 2017, Cai *et al.* proposed Cascade R-CNN [5]. This algorithm cascaded multiple parallel detection heads with an intersection over union (IOU) threshold, producing excellent expressive capabilities. However, the large receptive field of deep features in the network makes it difficult to extract the deep features of small objects, and poor feature information fusion in the pyramid network leads to poor overall object detection quality.

In this study, we propose an improved Cascade R-CNN aerial object detection algorithm, named RFNet. The main contributions of this paper are as follows:

- First, based on previous research, we optimized the feature pyramid network, and designed a better performing module named Recursive Feature Pyramid Network (RFPN), which is easy to implant into other networks.
- Second, we searched a recursive convolution kernel with an appropriate receptive field, and reinforced the method of feature map fusion by introducing a lightweight module to connect the feature semantic information.
- Finally, we improved the quality of object detection in aerial images by adding residual network (ResNet) [6] with an attention mechanism block (AMB) [7], and optimized the parameter of AMB to strengthen the ability of the CNN to extract features from the original image.

II. RELATED WORK

A. Up-sampling Module

The up-sampling module, one of the most widely used in various network structures, is an important part of a modern convolution network. One typical up-sampling method to improve spatial resolution is interpolating sub-pixel values based on the relationship between adjacent pixels in the previous image. Nearest neighbor interpolation and bilinear interpolation approximate the original image using functions of invariant space such as B-spline [8] and Hermite [9]. By calculating the missing pixels by formulas based on the value of surrounding pixels, researchers proposed guided adaptive interpolation al-

978-1-6654-0692-5/21/$31.00 ©2021 IEEE
DOI 10.1109/CISAI54367.2021.00020

gorithms [10-12] to overcome edge blur. The development of deep learning improved up-sampling [13-17] and performed significantly better than traditional methods.

B. Feature Pyramid Network

In a CNN, convolution and pooling operations are performed on the original image to obtain feature maps of different levels and sizes. A shallow network focuses on detailed information, whereas a deep network pays attention to semantic information, which can benefit the accurate detection of objects. The feature maps used in each prediction layer are detected with the corresponding resolution, ensuring that each layer has the appropriate resolution and strong semantic features.

To explore high-quality object detection algorithms and improve the speed and accuracy of real-time detection, many excellent detection models have been proposed. In 2015, Ren *et al.* proposed two-stage Faster R-CNN [3], and Redmon *et al.* proposed the one-stage you-only-look-once (YOLO) [18] object detection algorithm, using the feature map on the final convolution layer for prediction. In 2016, Liu *et al.* proposed the single shot multibox detector (SSD) [19], which unlike the previous algorithm, SSD extracts multi-scale feature maps for detection. In 2017, Lin *et al.* proposed the FPN [20] for object detection, which significantly improves detection performance for small objects, and many effective networks achieve high quality using the pyramid network. The top level feature map is fused with low level feature maps by up-sampling, and multi-stage feature maps are detected and regressed [21-24]. This significantly improves the quality of object detection by introducing a small amount of computation. In 2018, Liu *et al*. proposed the path aggregation network (PANet) [25], a top-down fusion method based on FPN, by expanding the bottom-up pathway fusion strategy. In 2020, Qiao *et al.* introduced recursion into the FPN and recursively fused the generated feature maps into the backbone [26].

III. Proposed Method

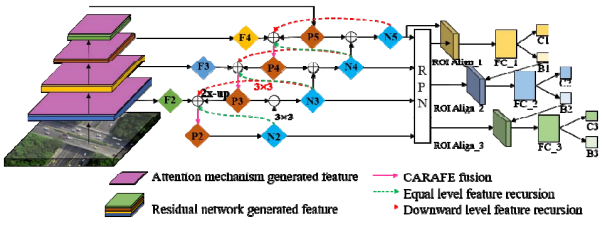

Fig. 1. Overall architecture of RFNet

As shown in Fig.1, RFNet primarily composed of three parts. The first part is the basic network, using ResNet-50 for the backbone network to extract feature, with an attention mechanism [7] introduced into each convolutional block to generate feature maps. For the second part, we designed recursive cross fusion strategy with a content-aware fusion algorithm. The third part retains the cascade multi-detector advantages of R-CNN [5].

A. Attention Mechanism

The most commonly used attention mechanisms in image processing are channel attention and spatial attention. In our method, we add attention modules in the conv2, conv3, conv4, and conv5 blocks. The feature map F ∈ C×H×W is set to global average pooling and max pooling. We obtain two feature maps M_C∈R^(C×1×1). The number of neurons in the first layer is C/r, r is equal to 16, and in the second layer is C, assuming V= [v_1, v_2, …, v_c]. This indicates the set of convolution kernels, where v_c represents the parameters of the C-th convolution kernel, and the output is U=[u_1, u_2, …, u_c]. The calculation formula is as follows:

$$u_c=\sum_{x=1}^{c} v_c^s * x^s \tag{1}$$

$$M_c(F)=\partial(MLP(AvgPool(F))+MLP(MaxPool(F))) \tag{2}$$

where $*$ represents the convolution operation. $M_C(F)$ is the output of the attention channel.

$$MLP(AvgPool(F))=W_1(W_2(F_{avg}^c)) \tag{3}$$

$$MLP(MaxPool(F))=W_1(W_2(F_{\max}^c)) \tag{4}$$

where F_{avg}^c and $F_{\max}^c$ represent the average pooling and max pooling feature maps, respectively, W_1 and W_2 are the weights of the two layers in the multilayer perceptron (MLP). The feature maps input to the spatial attention mechanism is F', where $F'=M_C(F)\otimes F$, output through a 7×7 convolution kernel, ∂ represents the sigmoid activation function, $\otimes$ is the Hadamard product, expressed as $M_S=\partial(f^{7\times7}(F_S))$, and the corresponding elements of the feature maps before entering the spatial attention mechanism are multiplied to output $F''=M_S(F')\otimes F'$.

B. Multi-scale Features

Multi-scale features are conducive to detecting objects of different sizes, and they effectively improve the detection effect of the detection model in practical applications. RFPN is optimized based on previous work. Specifically, the path from the bottom up that recurses to the top down, with a recursive convolution kernel size 1×1 and stride 1, is redesigned with a kernel size 3×3 and stride 1. The larger the convolution kernel, the larger the receptive field, and the more information that can be perceived, the better the global features that can be obtained. Furthermore, the last feature map generated is superimposed and fused again in the output update module.

The RFPN structure is shown in the second part of the overall architecture of RFNet in Fig. 1. The bottom-up path on the left (N2 to N5, named panet pathway), the feedforward pathway of the pyramid network, is calculated by feature mapping of different proportions. The feature map comprises step 2 ResNet blocks Conv1, Res_2, Res_3, Res_4, and Res_5, with the stride set to 2, 4, 8, 16, and 32, respectively, and the output feature map is {C2, C3, C4, C5}. The top-down path in the middle (P5 to P2, named fpn pathway) expands the resolution

of additional semantic high-level feature maps to generate high-resolution features. Through the content perception module, the feature space is reorganized and mapped to the upper adjacent feature space [16], and the feature is then mapped to the panet path through the result of a 1×1 convolution transformation. This is the same as the up-sampling result of the upper layer convolution, which then uses 3×3 for the merged result {M2, M3, M4, M5}, to the panet pathway, where the convolutional kernel size is 3×3 and the padding is 1. We introduce the recursive pathway as the fpn pathway to the pafpn pathway using a 3×3 kernel to eliminate overlapping effects, called the recursive pathway. In addition, to be consistent with the channels, input size, and output size, we use a 3×3 kernel recursive to the fpn pathway to enhance and extract the feature. The end result is a set of scale feature maps {P2, P3, P4, P5}, which are similar to {C2, C3, C4, C5}, as they have the same size and channels.

The RFPN structure retains the multi-scale pathway fusion of the typical feature pyramid network structure and adds features to the path for loop fusion. Based on lateral connection output multi-stage feature maps, $\forall i \in [1,S]$,

$$b_i^t = B_i^t(x_{i-1}) \tag{5}$$

we introduced the number of loop t, $\forall t \in [0,T]$,

$$f_i^t = F_i^t(f_{i+1}^t, B_i^t, R_i^t, L_i^t) \tag{6}$$

when $t=0$, there is no feature map generated recursively to the top-down pyramid, obtaining a network equivalent to PANet [23]. For any t, feedback R_i and L_i to the top-down path, and fusion with b_i , if $t=T$, p_i outputs the feature maps through the convolution operation,

$$p_i^t = P_i^t(p_i^t, f_i^t) \tag{7}$$

The output indicates that $O_i = p_i^T$. Note that R_i and L_i represents the feature map recursion from pyramid, as shown in Fig. 1, where P_i stands for the feature maps in the bottom-up pathway fusion. In particular, x_0 is the original image input, and the set effective value limit,

$$R_{S+1}^t = f_{S+1}^t = 0 \tag{8}$$

C. Feature Fusion Update

We redesigned the feature fusion update module mainly to include the feature map generated by the lateral connection, generated by the last loop (valid when $i<S$), generated by the content perception fusion of the $i+1$ layer feature, and generated recursively by the output. The feature fusion block update module has two main improvements over FPN. The 2x up-sampling method is replaced by a feature fusion method based on content perception. The carafe [16] network is mainly composed of two parts: a kernel prediction module, which is used to generate the weight on the core for reorganization calculation, and a content-aware reorganization module, a light-weight feature fusion used to convolute the calculated weight and reshape the channel into a $k \times k$ matrix as the core, with the corresponding points on the original input feature map and the $k \times k$ region with its center point to obtain the output.

IV. Experiment

A. Implementation Details

Our experiment was performed on the VisDrone [1] dataset. All the models presented in the paper were trained and verified based on 6471 and 548 labeled images, and we tested on 1610 labeled images from the dataset. We implemented the RFNet and the experiment on mmdetection [27]. The training process was done on Ubuntu18.04LTS. Our basic model used Cascade R-CNN [5], and we strictly followed its experimental settings for the experiment, setting the training model to 12 epochs in the comparison experiment and warm-up [28] equal to 0.001, meaning the learning rate was multiplied by 0.1 after 8 and 12 epochs. Additionally, other training and testing settings are kept the same and no bells and whistles are used for them.

B. Experimental Results

We selected several modern object detection methods to compare to RFNet. To intuitively show the convergence of each algorithm in the training process, we tested the validation dataset and performed calculations and evaluations for each epoch. As shown in Fig. 2, RFNet had a faster convergence speed, and the convergence value at epoch 12 was significantly higher than other algorithms. Furthermore, we observed the detection results of the three pyramids equipped with the Cascade R-CNN [5]. At 2 to 6 epochs, the accuracy of FPN was not significantly different from the PANet and RFPN. Because of the lightweight computing requirements of FPN, it is used to calculate shallow features in the early stages. However, with the iteration of training times, the top-down structure of FPN could not fully extract high-level semantic features. The pyramid structure of PANet [23] and RFPN were more advantageous in this respect. Therefore, in the later stages of training, PANet was significantly better than FPN [20]. RFNet added a recursive fusion strategy and introduced a way to focus on content information fusion, and the results observed for our method were significantly better than those of FPN and PANet in the range from 8 to 12 epochs.

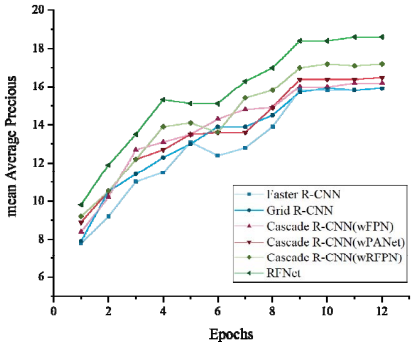

Fig. 2. Results of detection algorithms and comparison of training process

Fig. 3. RFNet algorithm detection results on the VisDrone dataset.

TABLE I: DETECTION RESULTS ON VISDRONE TEST DATASET.

Methods	Input Size	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	Params(M)
RetinaNet [21]	800×600	8.9	15.8	9.1	1.7	14.6	27.2	36.29
Faster R-CNN [3]	800×600	13.5	23.4	14.3	4.2	23.7	33.6	41.17
Grid R-CNN [30]	800×600	13.7	22.9	14.7	4.2	23.8	34.1	64.25
DetectoRS-50 [26]	~	13.99	-	-	-	-	-	-
Cascade R-CNN [5]	800×600	14.1	23.3	15.3	4.3	24.4	36.3	68.95
YOLOv4 [23]	~	15.47	-	-	-	-	-	-
RFNet	800×600	15.2	25.2	16.3	5.0	26.5	38.1	79.57
RFNet	850×650	16.0	26.3	17.2	5.7	27.3	40.3	79.57

Note: The detection results of DetectoRS-50 and YOLOv4 are from the announcement of competition results 2020. Other algorithms have been tested by the unified experimental details in this paper.

TABLE II: ABLATION STUDY ON VISDRONE VALIDATION DATASET.

RFPN	AM	Size	mAP	pedestrian	car	tricycle	motor	people	van	a-tricycle	bicycle	truck	bus
			16.2	8.6	41.8	10.9	9.8	4.3	25.4	6.4	3.8	20.4	31.3
√			17.2	9.0	42.8	12.0	10.1	4.5	27.0	7.7	4.2	21.9	32.9
√	√		17.5	8.8	42.9	12.3	10.3	4.9	27.3	8.0	4.6	22.8	32.9
√	√	√	18.5	10.3	44.6	12.7	12.1	5.4	28.4	8.7	5.1	23.0	34.4

Following the MS COCO [29] dataset detection standard, we used mAP, AP50, AP75, APS, APM, and APL indicators to evaluate the object detection algorithm under different IOUs and object sizes. As Table I shows, we calculated the model parameters that were used to estimate approximately the size of the model. It can be seen that RFNet algorithm has more parameters than the other algorithms, so in deployment process, the hardware platform requirements are slightly higher, but the accuracy is significantly higher than that of the other algorithms according to various indicators, showing it is suitable for applications with higher accuracy requirements. Moreover, to ensure that other conditions were the same, the input size had to be increased from 800×600 to 850×650, which increased the detection accuracy by 0.8%.

To better explain the contribution of each module in RFNet, we conducted ablation experiments on the VisDrone [1] validation dataset, including using RFPN, the attention mechanism block (AMB), and the visual display of the impact of size expansion on the dataset. As Table II shows, at the basic input size of 800×600 pixels, the original framework using Cascade R-CNN [5] with FPN obtained 16.2% mean average precision (mAP). Adding the designed RFPN module increased the precision by 1.0%, with the accuracy of the three categories of "van," "truck," and "bus" significantly improved by 1.6%, 1.5%, and 1.6%, respectively. Upon introducing the attention mechanism, a further increase of 0.3% was observed. When the size was expanded to 850×650, the mAP increased another 1.0%.

Finally, we randomly selected some representative detection results during the test, as shown in Fig. 3, including high-altitude, low-altitude, daytime, and night scenes and multi-angle detection effects. The experimental results show that our method in the aerial image experiment can achieve a high quality of object detection, and it has particular theoretical research and application value. One of our next tasks will be to expand the method to meet the needs of practical applications.

V. Conclusion

In this study, we proposed a deep convolutional neural network architecture, RFNet, to enhance the poor detection quality caused by insufficient feature fusion for aerial image detection. This included the design of the RFPN module, which recurses the features of the top-down pathway enhancing information flow, and a lightweight module to connect feature semantic information to replace up-sampling. In addition, we introduced an attention mechanism block into ResNet to efficiently extract the features. The RFPN module is easy to merge into other networks to improve detection quality.

Increased in the model parameters of RFNet was observed which resulted in an improvement in detection accuracy. However, the increase present associated limitations in computing conditions. Going forward, studies to improve detection accuracy while compressing the model parameters will be paramount.

Acknowledgment

This work was supported by the Robot Technology Used for Special Environment Key Laboratory of Sichuan Province Funded Project (13ZXTK07).

References

[1] D. Du, L. Wen, P. Zhu, H. Fan, and Z. Liu, "VisDrone-DET2020: The Vision Meets Drone Object Detection in Image Challenge Results," IEEE, 2020.

[2] R. Girshick, J. Donahue, T. Darrell, et al., "Rich feature hierarchies for accurate object detection and semantic segmentation," in Proc. IEEE Conf. Comput. Vision Pattern Recognit., 2014, pp. 580–587.

[3] S. Ren, K. He, R. Girshick, et al., "Faster R-CNN: towards real-time object detection with region proposal networks," in IEEE Trans. Pattern Anal. Mach. Intell., 2016, pp. 1137–1149.

[4] J. R. R. Uijlings, K. E. A. Van De Sande, T. Gevers, et al., "Selective search for object recognition," in Int. J. Comput. Vision, 2013, pp. 154–171.

[5] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in Proc. IEEE Conf. Comput. Vision Pattern Recognit., 2018, pp. 6154–6162.

[6] K. He, X. Zhang, S. Ren, et al., "Deep residual learning for image recognition," in Proc. IEEE Conf. Comput. Vision Pattern Recognit., 2016, pp. 770–778.

[7] S. Woo, J. Park, J. Y. Lee, et al., "Cbam: Convolutional block attention module," in Proc. Eur. Conf. Comput. Vision, 2018, pp. 3–19.

[8] P. H. C. Eilers and B. D. Marx, "Flexible smoothing with B-splines and penalties," in Statistical Sci, 1996, pp. 89–121.

[9] T. Kohonen, "The self-organizing map," in Proc. IEEE, 1990, pp. 1464–1480.

[10] L. Zhang and X. Wu, "An edge-guided image interpolation algorithm via directional filtering and data fusion," in IEEE Trans. Image Process., 2006, pp. 2226–2238.

[11] M. Li and T. Q. Nguyen, "Markov random field model-based edge-directed image interpolation," in IEEE Trans. Image Process., 2008, pp. 1121–1128.

[12] C. Arcelli, N. Brancati, M. Frucci, et al., "A fully automatic one-scan adaptive zooming algorithm for color images," in Signal Process., 2011, pp. 61–71.

[13] W. Shi, J. Caballero, F. Huszár, et al., "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in Proc. IEEE Conf. Comput. Vision Pattern Recognit., 2016, pp. 1874–1883.

[14] Z. Tian, T. He, C. Shen, et al., "Decoders matter for semantic segmentation: Data-dependent decoding enables flexible feature aggregation," in Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit., 2019, pp. 3126–3135.

[15] X. Hu, H. Mu, X. Zhang, et al., "Meta-SR: A magnification-arbitrary network for super-resolution," in Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit., 2019, pp. 1575–1584.

[16] J. Wang, K. Chen, R. Xu, et al., "Carafe: Content-aware reassembly of features," in Proc. IEEE/CVF Int. Conf. Comput. Vision., 2019, pp. 3007–3016.

[17] X. Ying, Y. Wang, L. Wang, et al., "A stereo attention module for stereo image super-resolution," in IEEE Signal Process. Lett., 2020, pp. 496–500.

[18] J. Redmon, S. Divvala, R. Girshick, et al., "You only look once: Unified, real-time object detection," in Proc. IEEE Conf. Comput. Vision Pattern Recognit., 2016, pp. 779–788.

[19] W. Liu, D. Anguelov, D. Erhan, et al., "Ssd: Single shot multibox detector," in Eur. Conf. Comput. Vision., 2016, pp. 21–37.

[20] T. Y. Lin, P. Dollár, R. Girshick, et al., "Feature pyramid networks for object detection," in Proc. IEEE Conf. Comput. Vision Pattern Recognit., 2017, pp. 2117–2125.

[21] T. Y. Lin, P. Goyal, R. Girshick, et al., "Focal loss for dense object detection," in Proc. IEEE Int. Conf. Comput. Vision, 2017, pp. 2980–2988.

[22] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," in arXiv preprint, 2018, arXiv:1804.02767.

[23] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," in arXivpreprint, 2020, arXiv:2004.10934.

[24] Q. Zhao, T. Sheng, Y. Wang, et al., "M2det: A single-shot object detector based on multi-level feature pyramid network," in Proc. AAAI Conf. Artif. Intell., 2019, pp. 9259–9266.

[25] S. Liu, L. Qi, H. Qin, et al., "Path aggregation network for instance segmentation," in Proc. IEEE Conf. Comput. Vision Pattern Recognit., 2018, pp. 8759–8768.

[26] S. Qiao, L. C. Chen, and A. Yuille, "Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution," in arXiv preprint, 2020, arXiv:2006.02334.

[27] K. Chen, J. Wang, J. Pang, et al., "MMDetection: Open mmlab detection toolbox and benchmark," in arXiv preprint, 2019, arXiv:1906.07155.

[28] P. Goyal, P. Dollár, R. Girshick, et al., "Accurate, large minibatch sgd: Training imagenet in 1 hour," in arXiv preprint, 2017 arXiv:1706.02677.

[29] T. Y. Lin, M. Maire, S. Belongie, et al., "Microsoft coco: Common objects in context," in Eur. Conf. Comput. Vision, 2014, pp. 740–755.

[30] X. Lu, B. Li, Y. Yue, et al., "Grid r-cnn," in Proc. IEEE/CVF Conf. on Comput. Vision Pattern Recognit., 2019, pp. 7363–7372.