the backbone network, and down-sampling technology is applied to the first three stages of the backbone network. After the fourth stage [21], maxpool of 2×2 is used, and the step size is set to 1, so that the resolution of the feature map of the fifth stage remains unchanged and the details of the image are retained. Therefore, there are only 3 down-sampling operations in the network, and the resolution is reduced by 1/8. In addition, in order to solve the problem that the receptive field is limited after the down-sampling operation is removed, the extended convolution technology is used in the convolution operation of the fifth stage, and the extended parameter is set to 2 to increase the receptive field of the model under the condition that the network parameters remain unchanged.

A backbone network based on CBAM attention

The new model introduces CBAM attention module in the backbone network. SENet (Squeeze-and-Excitation Networks) learns feature weights based on loss, so that the effective feature maps have high weights, and the ineffective/inefficient feature maps have low weights, so that the model can be improved for better results. However, the shortcoming of SENet only considers the importance of pixels in different channels, ignoring the importance of different positions [22-25]. CBAM model has more spatial attention mechanism than SENet. This spatial attention module can learn the importance of different positions of each feature map. The CBAM module structure is shown in figure 2.

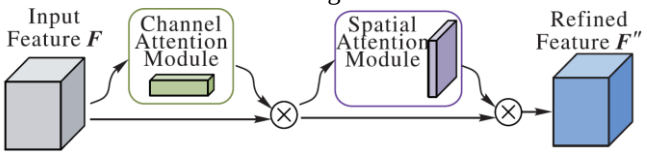


Figure 2. Module structure of CBAM

CBAM module can be divided into channel attention module (figure 3) and spatial attention module (figure 4). Specifically, in the channel attention module, average pooling and maximum pooling are used to compress the input feature graph F in the spatial dimension, and two different spatial descriptors F_{avg}^{c} and $F_{\max}^{c}$ are obtained, representing average pooling feature and maximum pooling feature respectively. Then, the two descriptors are sent to the multi-layer perceptron (MLP) [26], and the features of MLP output are added based on element-wise, and then activated by sigmoid to generate channel attention graph M_c. Finally, the feature graph F' with channel concern is obtained by multiplying F and M_c.

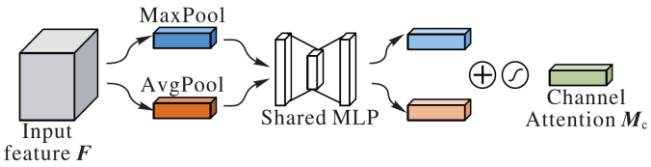


Figure 3. Channel attention module

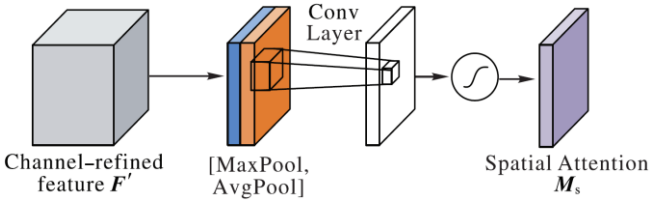


Figure 4. Spatial attention module

In the spatial attention module Stage1, average pooling and maximum pooling operations are first applied to F' along the channel dimension to obtain F_{avg}^{s} and $F_{\max}^{s}$. Then, these two features are connected and convolved through the standard convolution layer, and then activated by sigmoid to generate spatial concern graph M_s. Finally, the spatial attention graph M_s is multiplied by F' to obtain the feature graph F'' with attention mechanism.

Considering the problem of network parameters, only part of the network convolution layer is added with CBAM attention mechanism, as shown in Figure 1. After the CBAM module is added, the network can learn the importance of different feature graphs and pixels at different positions, thus enhancing the feature extraction ability of the model.

Max pooling

Modern convolutional networks are not shift-invariant, because commonly used down-sampling methods, such as Maxpool strided-Conv and Avgpool, ignore the sampling theorem, so a small input shift or translation will lead to dramatic changes in the output. To solve this problem, Chintala et al. [27] proposed a BlurPool sampling method. As shown in figure 5, the first step in maximum pooling is to calculate the maximum value of the region and then perform down-sampling. BlurPool, on the other hand, inserts the anti-aliasing operation in the middle and smoothes the input signal by introducing a blur core so that the translated result is similar to the untranslated result.

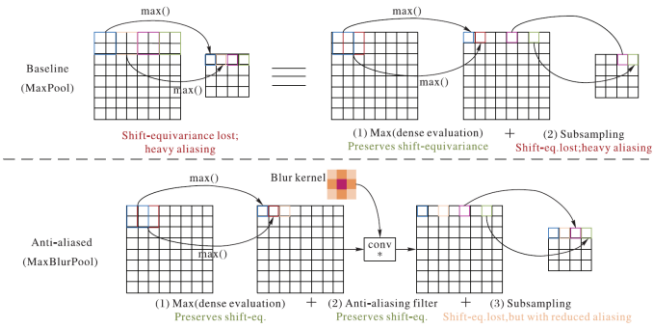


Figure 5. Anti-aliased max-pooling

Take one-dimensional signal as an example, as shown in figure 6, when the input signal is [0,0,1,1,0,0,1,1], traditional maximum pooling will result in [0,1,0,1]. If the input signal is shifted by 1 step, the maximum pooling will be [1,1,1,1]. The two results vary greatly and the serration is obvious. And MaxBlurPool operation to do the input signal is first step 1 for maximum operating [0,1,1,1,0,1,1,1], ends to vacate get [1,0,1,1,1,0,1,1,1,0], then introduce the fuzzy core kernel=[1, 2, 1], the fuzzy kernel and signal to do the dot product operation, and divided by the fuzzy of nucleus and value, get [0.5,0.75,1,0.75,0.5,0.75,1,0.75], redo the sample after get [0.5,1,0.5,1]. Similarly, if the input signal is shifted by 1 step, the final result will be [0.75,0.75, 0.75,0.75,0.75,0.75], which is relatively smoother than the traditional maximum pooling. It should be noted that multiple fuzzy cores are provided in reference [28], and manual selection is required when using fuzzy kernels.

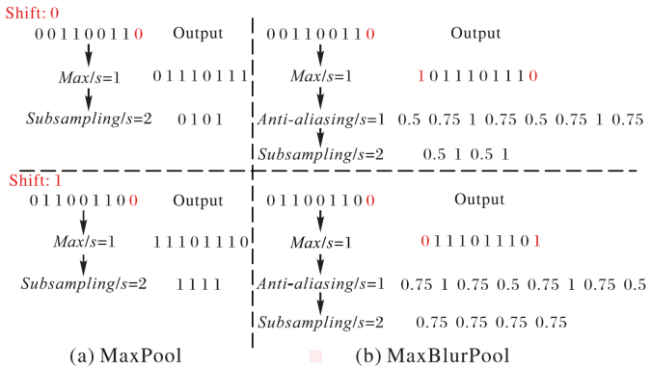


Figure 6. Anti-aliasing operations for one-dimensional signals

The new model uses this down-sampling technique after the first three phases of the network to enhance the robustness and generalization of the model.

Multi-scale feature extraction

In deep learning, there are usually two ways for the network to learn multi-scale features: the first method is inside the neural network by increasing the receptive field and sampling layer in the network, so that the features learned at each layer are naturally multi-scale; The second method is by adjusting the size of the input image.

CFF extracts multi-scale features of images in the same way as RCF network. The side outputs of each layer in the network of the trunk are characteristic compressed through 1×1 convolution layer, and all side outputs are added up in the unit of stage. Then, through 1×1 convolution layer dimensionality reduction, a feature graph of a single channel is output.

Feature pyramid fusion module

In order to fully integrate the multi-scale features of each level, the new model adopts the feature pyramid method, which makes the lower level also pay attention to the global features by transferring the features of the higher level to the lower level. While fully integrating multi-scale features, the problem of fuzzy details in low-stage generated feature maps is solved effectively.

The Feature Fusion Module (FFM) in figure 1 firstly up-samples the features of the upper layer, and then connects them with the features of the lower layer. Then Feature compression is carried out through the convolution of 1x layer. In this way, the lower level is able to integrate the features of the higher level. In addition, in order to avoid over-ignoring the important details of low-level features, the residual network structure is used for reference. In this module, the original feature graph and the output feature graph are added together, and the result is taken as the final output of FFM. The FFM structure in stage 4 is shown in figure 7. The structure is similar in other layers.

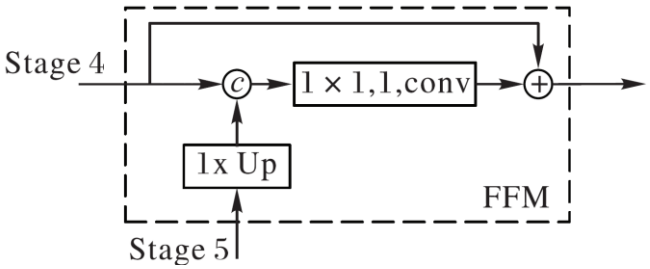


Figure 7. Feature fusion module

For the output of the feature fusion module: on the one hand, it is the input of the fusion module in the previous stage; On the other hand, deconvolution (deconv)

operation is used to realize up-sampling, so that each stage outputs an edge graph, and the model carries out supervised learning of the edge graph output by each stage.

2.3. Contour detection

The edge image is obtained by edge detection and non-maximum suppression, and the threshold control is used to filter out noise and false edge caused by small change, so as to obtain accurate contour image. In this paper, the Otsu method is used to further segment the robot edge image. The segmentation value of the maximum inter-class variance is taken as the threshold value, and the edge detection results are tested twice to obtain the final contour, so as to realize the adaptive edge detection. Where, the inter-class variance is defined as formula (1):

$$L = \omega_1 \times (\mu - \mu_1)^2 + \omega_2 (\mu - \mu_2)^2 \quad (1)$$

Where ω_1 and ω_2 are the proportion of target pixel and background pixel respectively. μ_1 and μ_2 are the average edge response intensity of target pixel and background pixel.

μ is the total average intensity of edge image, defined as formula (2):

$$\mu = \omega_1 \times \mu_1 + \omega_2 \times \mu_2 \quad (2)$$

2.4. Multi-scale feature fusion

After the feature pyramid module, multi-scale features have been fully fused, so the model only uses one 1×1 convolution layer to fuse multi-scale features of all levels, as the final output edge image of the new model, and carries out supervised learning.

2.5. Loss function

Edge detection datasets are typically marked by multiple annotators. For each image, all annotators' marks are averaged to generate edge probability plots ranging from 0 to 1. Where 0 indicates that no annotator marks the pixel; 1 means all annotators are marked at this pixel. Pixels with edge probability higher than η are regarded as positive samples, and pixels with edge probability equal to 0 are regarded as negative samples. Otherwise, if a pixel is marked by an annotator less than n, the pixel may be a semantically disputed edge point, and whether it is treated as a positive sample or a negative sample may confuse the network, so the model ignores pixels in this category.

Because edge detection is to classify pixels, this model uses cross entropy function as the objective function. Same as reference [29], the loss of each pixel relative to pixel label is calculated as:

$$l(X_i;W) = \begin{cases} \alpha \cdot \log(1 - P(X_i;W)), & y_i = 0 \\ 0, & 0 < y_i \le \eta \\ \beta \cdot \log P(X_i;W), & y_i > \eta \end{cases} \quad (3)$$

$$\alpha = \lambda \cdot \frac{|Y^+|}{|Y^+| + |Y^-|} \quad (4)$$

$$\beta = \frac{|Y^-|}{|Y^+| + |Y^-|} \quad (5)$$

Where $|Y^+|$ and $|Y^-|$ represent the number of positive and negative samples respectively. The hyperparameter λ is used to balance positive and negative sample size differences. X_i represents the activation value of the neural network and the probability value that pixel i in the label graph is the edge point. W represents the learnable parameter in the neural network.

2.6. Setting the weight of loss at different stages

In the network, the output edge images of each stage differ greatly, and the magnitude of loss of each stage may be inconsistent, and the loss of fusion stage should be dominant.

In addition, it is found in the experiment that when the model is trained to the 20th epoch, the feature maps of the first two stages almost no longer contains any detailed textures, which may be the negative effects of the integration of low-level features with high-level features. These problems are not good for the final forecast.

In order to restrain this phenomenon, this paper reduces the proportion of loss in the five stages of the network and increases the proportion of loss in the fusion stage, so as to balance the relationship between loss in each stage and loss in the fusion stage. The loss weight of the five stages of the network is set as S^{side}, and the final loss weight of the fusion layer is S^{fuse}, so the total loss function can be written as:

$$L(W) = \sum_{i=1}^{|I|} \left(\sum_{k=1}^{K} S^k \cdot l(X_i^{(k)};W) + s^{fuse} \cdot l(X_i^{fuse};W) \right) \quad (6)$$

Where S^k represents the loss weight of the k-th stage. s^{fuse} represents the loss weight of the fusion layer. $X_i^{(k)}$ represents the excitation value of the i-th pixel in the output image of the k-th stage. X_i^{fuse} represents the

excitation value of the i-th pixel in the image output by the fusion module. U represents the total number of pixels in each image, and K represents the number of stages in the backbone network.

2.7. Multi-scale edge detector

In order to further improve the quality of edge, image pyramid technology is adopted in the test. Specifically, an image pyramid is constructed by re-sizing images during testing, and each image is fed separately into a trained single-scale detector. Then, bilinear interpolation is used to adjust all the edge probability graphs to the size of the original image. Finally, the weighted average of these results is used to obtain the final predicted edge graph. This model uses three different scales, 0.5, 1.0 and 1.5 respectively.

3. Model training

The BSDS500 [30] dataset and PASCAL VOC Context200 dataset are widely used in edge detection. The BSDS500 dataset consists of 200 training images, 100 validation images and 200 test images, each labeled by 4 to 9 annotators. In order to prevent the over-fitting phenomenon of the model, rotation, expansion and clipping of 300 images in the training set and verification set of BSDS500 are carried out to enhance the data set. Finally, the enhanced data set of BSDS500 is mixed with PASCAL VOC Context data set as training data.

The new network is written based on Python3, using the pytorch 1.0.1 deep learning framework, and several other libraries. The experiment is carried out on an Ubuntu server with hardware including E5-2678 V3 2.50 GHz CPU and an NVIDIA TeslaK40C video card. The video memory 12 GB model is trained with 30 epochs by stochastic gradient descent algorithm. Batch size is set to 1, the benchmark learning rate is set to 1E-6, different learning rates are specified for different convolutional layers, momentum is set to 0.9, weight decay is set to 0.0002. During training, no pre-training model is used and network parameters are initialized using Gaussian distribution [31-35].

4. Experiments and analysis

Given an edge probability graph, a threshold is required to produce an edge image, and there are two options for setting this threshold. The first is the Optimal Dataset Scale (ODS), which applies a fixed threshold to all images in the dataset. The second is the Optimal Image Scale (OIS), which selects an Optimal threshold for each Image. ODS and OIS are commonly used as indicators of edge detection models.

4.1. Experiments analysis

The non-maximum suppression technique [9] is applied to the Edge image output by the model to obtain the refined edge image for evaluation, and the Edge Box tool kit is used for evaluation. Figure 8 shows the evaluation results. Compared with traditional methods, the edge detection of RCF network has achieved better results, and the CFF model optimizes the shortcomings of RCF network, and its multi-scale strategy improves the ODS score to 0.818.

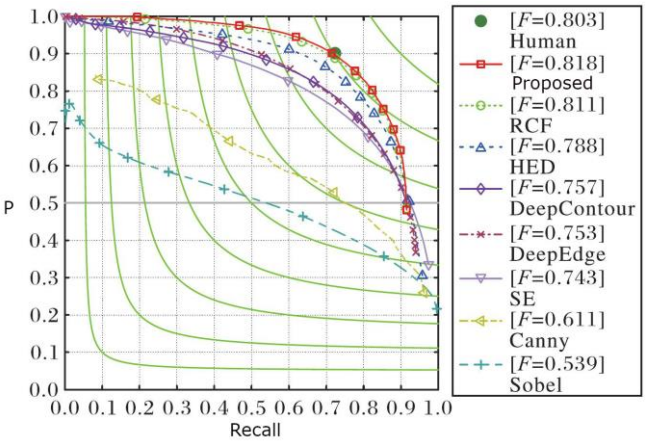


Figure 8. Evaluation results on BSDS500

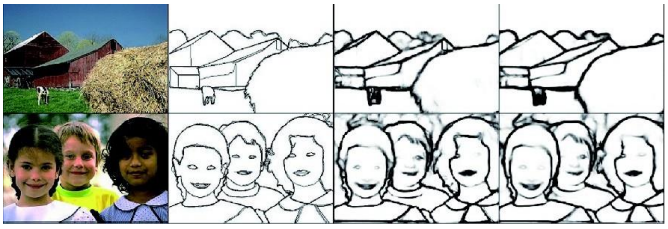
Proposed method is compared with other related algorithms, and the results are shown in table 1. As can be seen from the indicators in the table, the ODS and OIS of the proposed model are 0.7% and 0.9% higher than the RCF model, respectively.

Table 1. Comparison between proposed model and other algorithms

Method	ODS	OIS
Canny	0.622	0.687
Gpb-ucm	0.737	0.768
SE	0.754	0.774
N4-fields	0.764	0.780
DeepEdge	0.764	0.783
DeepContour	0.767	0.784

HED	0.799	0.815
RCF	0.822	0.841
Proposed	0.829	0.850

The comparison results of edge images output by new method and RCF networks are shown in figure 9. It can be seen from the comparison that some lines in the edge image generated by RCF model are fuzzy, while the new model can clearly detect the edges in the image and deal with some fuzzy details better.



Original (b) ground truth (c) RCF (d) proposed

Figure 9. Comparison of the results of proposed and RCF

The comparison between the proposed model and the edge image output by RCF network is shown in figure 9. It can be seen from the comparison that some lines in the edge image generated by RCF model are fuzzy, while the new model can clearly detect the edges in the image and deal with some fuzzy details better.

In order to further demonstrate the optimization details of the proposed model, the comparison between the edge images output by the proposed model at each stage and the RCF network is given in figure 10. In the figure, each column is the edge image generated from stage 1-5 from top to bottom. It can be seen that each stage of RCF network has poor processing ability for some irrelevant details, and each stage contains some fuzzy lines. The proposed model can focus on some global contour information in the lower layer by integrating features of different levels across layers, which helps to fully integrate multi-scale features. As can be seen from the figure, the edge image output by the proposed model only contains few irrelevant details compared with RCF, especially in the first and second stages, without too much messy texture.

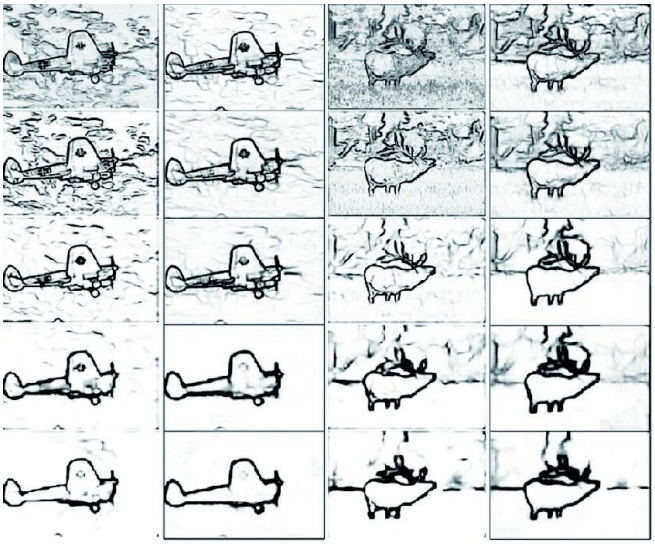


Figure 10. Comparison of output images of CFF and RCF at each stage. From fist row to fifth row: stage1-stage5. From fist column to fifth column: RCF (airplane), proposed (airplane), RCF (wapiti), proposed (wapiti).

4.2. Ablation experiments

In this section, the internal structure of the proposed model is analyzed. As shown in Table 2, after the introduction of CBAM attention module and anti-aliasing downsampling technology into the trunk network, the ODS and OIS of the model increases by 0.4% and 0.5%, proving that the feature information extracted from the trunk network of this model is more abundant and effective. In addition, both ODS and OIS improve by a further 0.2% fusing the output features of different stages across layers, indicating that multi-scale features can be fully fused by transferring high-level features to low-level features. In order to balance the relationship between losses at different stages and suppress the loss of low-level details, the ODS and OIS of the model were increased by 0.1% and 0.2% respectively by setting the weight of losses at different stages.

Table 2. Comparison of improvement effect of different modules

No.	CBAM+ Dilation+ MaxBlurPool	FFM	Weight loss	ODS	OIS

1				0.822	0.841
2	√			0.826	0.846
3	√	√		0.828	0.848
4	√	√	√	0.829	0.850

5. Conclusion

This paper proposes a global edge detection network based on RCF network. In the new model, the CBAM module is added into the VGG16 backbone network and the down-sampling technique with translation invariance is adopted to improve the feature extraction capability of the network. Part of the lower sampling layer is removed to prevent the image resolution from being too low and affecting the model accuracy. At the fifth stage, dilated convolution technology is used to improve the receptive field of the network. In addition, the model adopts a feature fusion mode from depth to shallowness, which makes the network pay more attention to the global information, and sets different weight of loss for different stages to balance the loss of each stage, preventing the model from excessively ignoring the details of the lower layer. Experiments show that the new model can generate clearer edge images.

Acknowledgements.

The author greatly appreciates the reviewers' anonymous comments.

References

[1] Tang Z, Chen Y, Ye S, et al. Fully Memristive Spiking-Neuron Learning Framework and Its Applications on Pattern Recognition and Edge Detection[J]. Neurocomputing, 2020, 403.

[2] Shoulin Yin, Hang Li, Asif Ali Laghari, et al. A Bagging Strategy-Based Kernel Extreme Learning Machine for Complex Network Intrusion Detection[J]. EAI Endorsed Transactions on Scalable Information Systems. 21(33), e8, 2021. http://dx.doi.org/10.4108/eai.6-10-2021.171247

[3] Shoulin Yin, Ye Zhang and Shahid Karim. Region search based on hybrid convolutional neural network in optical remote sensing images[J]. International Journal of Distributed Sensor Networks, Vol. 15, No. 5, 2019. DOI: 10.1177/1550147719852036

[4] Lin Teng, Hang Li, Shoulin Yin, Yang Sun. Improved krill group-based region growing algorithm for image segmentation[J]. International Journal of Image and Data Fusion. 10(4), pp. 327-341, 2019. doi: 10.1080/19479832.2019.1604574

[5] Shoulin Yin, Ye Zhang, Shahid Karim. Large Scale Remote Sensing Image Segmentation Based on Fuzzy Region Competition and Gaussian Mixture Model[J]. IEEE Access. volume 6, pp: 26069 - 26080, 2018.

[6] Jing M, Du Y. Flank Angle Measurement Based on Improved Sobel Operator[J]. Manufacturing Letters, 2020, 25(5).

[7] Zhang L, Zou L, Wu C, et al. Method of famous tea sprout identification and segmentation based on improved watershed algorithm[J]. Computers and Electronics in Agriculture, 2021, 184(1):106108.

[8] Jamsawang P. Learning to detect natural image boundaries using local brightness, color, and texture cues. 2018.

[9] P. Dollár and C. L. Zitnick, "Fast Edge Detection Using Structured Forests," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 8, pp. 1558-1570, 1 Aug. 2015, doi: 10.1109/TPAMI.2014.2377715.

[10] Ganin Y, Lempitsky V. $ N^4 $-Fields: Neural Network Nearest Neighbor Fields for Image Transforms[J]. Asian Conference on Computer Vision, 2014.

[11] Xie S, Tu Z. Holistically-Nested Edge Detection[J]. International Journal of Computer Vision, 2015, 125(1-3):3-18.

[12] Qingwu Shi, Shoulin Yin, Kun Wang, et al. Multichannel convolutional neural network-based fuzzy active contour model for medical image segmentation. Evolving Systems (2021). https://doi.org/10.1007/s12530-021-09392-3

[13] Ting-Ting Gao, Hang Li, and Shou-Lin Yin. Adaptive Convolutional Neural Network-based Information Fusion for Facial Expression Recognition [J]. International Journal of Electronics and Information Engineering. Vol. 13, No. 1, pp. 17-23, 2021.

[14] Y. Liu, M. Cheng, X. Hu, K. Wang and X. Bai, "Richer Convolutional Features for Edge Detection," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5872-5881, doi: 10.1109/CVPR.2017.622.

[15] Woo S., Park J., Lee JY., Kweon I.S. (2018) CBAM: Convolutional Block Attention Module. ECCV 2018. ECCV 2018. Lecture Notes in Computer Science, vol 11211. Springer, Cham. https://doi.org/10.1007/978-3-030-01234-2_1

[16] Zhang R. Making Convolutional Networks Shift-Invariant Again[C]\\ Proceedings of the 36th International Conference on Machine Learning. New York : International Machine Learning Society, 2019: 7324-7334.

[17] Chen Y, Zhao H, Hu Z, et al. Attention-based context aggregation network for monocular depth estimation[J].

International Journal of Machine Learning and Cybernetics, 2021(11):1-14.

[18] Tan Y M, Wu P, Zhou G, et al. Combining Residual Neural Networks and Feature Pyramid Networks to Estimate Poverty Using Multisource Remote Sensing Data[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2020, 13(99):553-565.

[19] Y. Liu, M. Cheng, X. Hu, K. Wang and X. Bai, "Richer Convolutional Features for Edge Detection," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5872-5881, doi: 10.1109/CVPR.2017.622.

[20] Xie S, Tu Z. Holistically-Nested Edge Detection[J]. International Journal of Computer Vision, 2015, 125(1-3):3-18.

[21] Jie Song, Yu Yu, Qifeng Luo. Cross-layer fusion feature based on richer convolutional features for edge detection [J]. Journal of Computer Applications, 2020, 40(7): 2053 - 2058. (In Chinese)

[22] Shoulin Yin, Hang L*, Desheng Liu and Shahid Karim. Active Contour Modal Based on Density-oriented BIRCH Clustering Method for Medical Image Segmentation [J]. Multimedia Tools and Applications. Vol. 79, pp. 31049-31068, 2020.

[23] S. Yin and H. Li. Hot Region Selection Based on Selective Search and Modified Fuzzy C-Means in Remote Sensing Images[J]. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 5862-5871, 2020, doi: 10.1109/JSTARS.2020.3025582.

[24] Shahid Karim, Ye Zhang, Shoulin Yin, Irfana Bibi. A Brief Review and Challenges of Object Detection in Optical Remote Sensing Imagery [J]. Multiagent and Grid Systems. 16(3), 227-243, 2020

[25] Jing Yu and Lulu Zhao. A Novel Deep CNN Method Based on Aesthetic Rule for User Preferential Images Recommendation[J]. Journal of Applied Science and Engineering. Volume 24, Issue 1, 2021.

[26] Campo F, Neri M, Villegas O, et al. Auto-adaptive multilayer perceptron for univariate time series classification[J]. Expert Systems with Applications, 2021, 181:115147.

[27] Chintala S, Ranzato M, Szlam A, et al. Scale-invariant learning and convolutional networks[J]. Applied and Computational Harmonic Analysis, 2017.

[28] Yin, S., Li, H. & Teng, L. Airport Detection Based on Improved Faster RCNN in Large Scale Remote Sensing Images [J]. Sensing and Imaging, vol. 21, 2020. https://doi.org/10.1007/s11220-020-00314-2

[29] Xiaowei Wang, Shoulin Yin, Hang Li. A Network Intrusion Detection Method Based on Deep Multi-scale Convolutional Neural Network[J]. International Journal of Wireless Information Networks. 27(4), 503-517, 2020.

[30] Y Wang, Wang L, J Qiu, et al. Feature enhancement: predict more detailed and crisper edges[J]. Signal, Image and Video Processing, 2021:1-8.

[31] Xiaowei Wang, Shoulin Yin, Desheng Liu, et al. Accurate playground localisation based on multi-feature extraction and cascade classifier in optical remote sensing images [J]. International Journal of Image and Data Fusion, vol. 11, no. 3. pp. 233-250, 2020. DOI: 10.1080/19479832.2020.1716862 (2020.1.15) EI(JA)

[32] Xiaowei Wang, Shoulin Yin, Ke Sun, et al. GKFC-CNN: Modified Gaussian Kernel Fuzzy C-means and Convolutional Neural Network for Apple Segmentation and Recognition [J]. Journal of Applied Science and Engineering, vol. 23, no. 3, pp. 555-561, 2020.

[33] Shoulin Yin, Hang Li, Lin Teng, et al. An optimised multi-scale fusion method for airport detection in large-scale optical remote sensing images [J]. International Journal of Image and Data Fusion, vol. 11, no. 2, pp. 201-214, 2020. DOI: 10.1080/19479832.2020.1727573 (2020.2) EI(JA)

[34] Yin, S., Li, H. GSAPSO-MQC:medical image encryption based on genetic simulated annealing particle swarm optimization and modified quantum chaos system. Evolutionary Intelligence (2020). doi: 10.1007/s12065-020-00440-6

[35] Karim, S., Zhang, Y., Yin, S. et al. Impact of compressed and down-scaled training images on vehicle detection in remote sensing imagery. Multimedia Tools and Applications 78, 32565–32583 (2019).