Single Shot MultiBox Detector-based Feature Fusion Model For Building Object Detection

Xiaoyan Zheng and Xiaoli Zhao*

Department of Civil Engineering and Architecture, Zhengzhou University of Science and Technology, Zhengzhou, 450064, China

**Corresponding author. E-mail: ancrum@qq.com*

Received: Feb. 26, 2024; Accepted: Mar. 25, 2024

Aiming at the problems of low detection accuracy and large size of existing building detection models, which lead to the imbalance between remote sensing image detection speed and accuracy and are not conducive to later deployment, a building detection method based on single shot multiBox detector (SSD)-based feature fusion model is proposed. In this method, the feature extraction module extracts the features from the input image, and the discrimination features and interference features are decomposed by the feature decomposition module. Finally, the identification features are input into the multi-scale detection module for target detection. The interference features removed after feature decomposition are unfavorable to target detection, including complex background clutter, while the retained identification features are favorable to target detection, including targets of interest, thus effectively reducing false alarms and missing alarms and improving the detection performance of building targets. Experiments show that the interaction ratio, accuracy and total accuracy of proposed method on WHU data set reach 94.2%, 97.0% and 98.9% respectively, showing good effectiveness without significantly increasing parameters.

Keywords: building detection; single shot multiBox detector; feature fusion; multi-scale detection

http://dx.doi.org/10.6180/jase.202502_28(2).0017

1. Introduction

As the main places of human production and life, buildings are one of the key elements of a city, and their detection has been widely used in urban planning, land supervision, digital city modeling, disaster assessment and other fields. With the rapid development of remote sensing technology, building detection based on remote sensing images has gradually become a research hotspot. Compared with optical images, the background of remote sensing images is complex and changeable, and the information is large. The traditional detection algorithm is easy to cause problems such as low precision, poor generalization ability, slow speed, and more manual intervention [1, 2]. Therefore, it is of great significance to study a fast and accurate building detection model of remote sensing image.

In recent years, deep learning-based object detection has developed rapidly, among which convolutional neural networks (CNNs) are commonly used, which can automatically extract effective features from image data and have strong generalization ability [3]. Thus, a series of target detection algorithms are generated, which can be roughly divided into two categories: One is the two-stage target detection algorithm represented by Region CNN [4], Fast RCNN [5], and Faster RCNN [6], whose main idea is to first generate regional candidate boxes and then input them to CNN for further classification; The other is the single-stage target detection algorithm represented by SSD (single shot multibox detector) [7] and YOLO (you only look once) series [8], which is an end-to-end target detection framework that can directly output the category of the target to be detected.

Based on the above two types of methods, many scholars have also proposed improved methods and applied them to remote sensing image building detection. For example, Liu et al. [9] improved the training efficiency of Faster RCNN by using the approximate joint training method, optimized the model parameters, and achieved higher accuracy and recall rate. Shen et al. [10] removed the feature map with a resolution of 13×13 in YOLOv3 and used K-means algorithm to cluster remote sensing building data sets, effectively improving the detection ability of small targets. Jang et al. [11] combined the fully convolutional neural network (FCN) with the residual network (ResNet50) to detect buildings, which could improve the extraction accuracy of building areas. Liu et al. [12] introduced the cascade structure into VGG-Net, designed a cascade fully convolutional neural network, and introduced dilated convolution at the same time to increase the receptive field and improve the detection accuracy. Based on the Faster RCNN algorithm, Ding et al. [13] adopted deformable convolution to improve the adaptability to collapsed buildings with arbitrary shapes, and proposed a new method to estimate the intersection proportion of objects to describe the intersection degree of boundary boxes, which could achieve better detection accuracy and recall rate. Bai et al. [14] used DRNet, ROI Align and texture information in Faster RCNN to solve the region mismatching problem. Hua et al. [15] reconstructed the backbone network of YOLOv3 according to the importance of features, introduced RBF module, optimized the anchor frame and distribution mode, and effectively improved the generalization ability of the model. Gaihua et al. [16] proposed a multi-scale group convolutional neural network with attention mechanism based on Mask RCNN network, which could effectively improve the detection performance of small targets.

The research focus of the above references is to improve the detection accuracy of the model, but the size of the model is not taken into account, which is not conducive to the later deployment. Model deployment should take into account not only detection speed and accuracy, but also model volume and computation. The existing model compression methods include channel pruning, knowledge distillation and quantization. In addition, there are some lightweight networks designed specifically for mobile terminals, such as SqueezeNet [17], which replaces the 3×3 convolution with the 1×1 convolution, officially developing the direction of model compression. The MobileNets [18] used deep separable convolution, which could greatly reduce the amount of computation and network parameters, but it could affect the detection accuracy of the model. On this basis, the improved MobileNetv2 and MobileNetv3 have smaller model volume and better detection accuracy. Tan et al. [19] proposed EfficientDet, which used the feature fusion module Bi-FPN with cross-scale connection and the joint scaling method, with fewer parameters and faster inference speed. Han et al. [20] proposed a new basic unit of neural network Ghost module in 2020, and then built a new lightweight neural network GhostNet. Some scholars have simplified the existing mature network structure. For example, YOLOV4-tiny is optimized and modified based on the network structure of YOLOv4. This algorithm has low requirements on hardware configuration and fast detection speed. However, due to the relatively simple network layer, the detection effect is slightly poor.

To sum up, in order to effectively balance the relationship between detection speed, detection accuracy and model volume, this paper proposes a single shot multiBox detector (SSD)-based feature fusion model. In this method, the feature extraction module extracts the features from the input image, and the discrimination features and interference features are decomposed by the feature decomposition module. Finally, the identification features are input into the multi-scale detection module for target detection.

2. Proposed building detection model

2.1. Overall framework of building detection model

The flow chart of the proposed method is shown in Fig. 1. This method is based on SSD object detection framework, which mainly consists of three modules: feature extraction module, feature decomposition module and multi-scale detection module. Considering that the original SSD network uses multi-scale feature map for target detection, and the first feature map participating in detection is the output feature map of conv4_3 layer. The method in this paper aims to decompose the input features into two parts: the identification features that are favorable to detection and the interference features that are unfavorable to detection, and only the identification features are used to complete the target detection, so as to reduce false alarms and missing alarms. Therefore, the method in this paper needs to complete feature decomposition before starting detection. In addition, the feature extraction module needs to have sufficient feature extraction capability to learn features with certain semantic information. Therefore, conv3_3 and previous convolution layers in the original SSD network are selected for initial feature extraction, and layers conv4_1 to conv4_3 are used for feature decomposition.

The detailed structure of the feature extraction module is shown in Table 1, where k, s, p, and n represent kernel size, step size, filling number, and number of cores

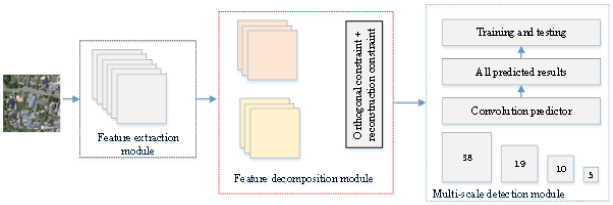

Fig. 1. The flow chart of presented building target detection

respectively. The module has a total of 7 convolutional layers, each of which is followed by Rectified Linear Unit (ReLU) [21] layers, and features are reduced by a maximum pooling layer after the 2th, 4th and 7th ReLU layers. The convolution kernel size of the 7 convolutional layers is 3×3, the convolution step size is 1, the number of convolutional kernel is 64, 64, 128, 128, 256, 256, 256, and the size of the pooling window of the three largest pooling layers is 2×2, and the pooling step size is 2. The feature extraction module extracts the initial feature of the input image for use in the subsequent feature decomposition module.

Table 1. Detailed structure of the feature extraction module

Layer	Operation	Hyper-parameter
1	Convolution, ReLU	k=(3,3), s=1, p=1, n=64
2	Convolution, ReLU	k=(3,3), s=1, p=1, n=64
3	Max-pooling	k=(2,2), s=2, p=0
4	Convolution, ReLU	k=(3,3), s=1, p=1, n=128
5	Convolution, ReLU	k=(3,3), s=1, p=1, n=128
6	Max-pooling	k=(2,2), s=2, p=0
7	Convolution, ReLU	k=(3,3), s=1, p=1, n=256
8	Convolution, ReLU	k=(3,3), s=1, p=1, n=256
9	Convolution, ReLU	k=(3,3), s=1, p=1, n=256
10	Max-pooling	k=(2,2), s=2, p=0

The feature decomposition module decomposes the input features. The module consists of two parallel branches, and the input features are decomposed into interference features and discrimination features by orthogonal constraints and reconstruction constraints between the two branches. The decomposed interference features are unfavorable to target detection, including complex background clutter. The discriminating feature is the part that is beneficial to the target detection, including the target of interest. As the output of the feature decomposition module, the discriminating features will be input to the subsequent multi-scale detection module for detection.

The multi-scale detection module performs target detection on the input identification features. The module consists of 13 convolutional layers, which are used to obtain convolutional feature maps of different scales from different layers of the network. In order to better detect the target, in addition to the discrimination feature map obtained by the feature decomposition module, convolutional feature maps of 5 scales obtained by the convolution layers of 5, 7, 9, 11 and 13 in the multi-scale detection module are selected. The size of these 6 feature maps gradually decreases from 39×39 to 1×1. Because the feature maps of different layers in the network have different receptive fields, that is, each position on the feature maps of different layers corresponds to the area of the original map. Among them, the receptive field corresponding to the low-level feature map is small, while the receptive field corresponding to the high-level feature map is larger, so the feature map of different layers can be used to perceive the target of different sizes. Selecting the multi-scale feature map for target detection is helpful to detect the target of various sizes in the original image. As shown by the multi-scale detection module surrounded by the green dashed line box in Figure 1, the multi-scale detection module also includes multiple convolutional predictors, each of which includes two convolution layers in parallel for border regression and classification.

2.2. Feature decomposition module

The input of the feature decomposition module is the feature map output of the feature extraction module, which aims at feature decomposition of the input feature map, removing the unfavorable part of the target detection, while retaining the favorable part of the target detection. The feature decomposition module consists of two parallel branches, namely interference feature extraction branch and discrimination feature extraction branch. The two branches have the same network structure and both contain three convolutional layers, each of which is followed by ReLU layer. The convolution kernel size of the three convolutional layers is 3×3, the convolutional step size is 1, and the kernel number of convolutional is 512. Since the parameters of the two branches are not shared, interference feature extraction and discrimination feature extraction are respectively undertaken. Finally, the obtained interference features and discrimination features are added and input to the decoder for image reconstruction. The decoder contains five deconvolution layers, in which the first four deconvolution layers are followed by ReLU layers. The convolution kernel size of the five deconvolution layers is 3×3, the convolution step size is 1, 2, 2, 2, 1, and the number of convolutional nuclei is 512, 256, 128, 64, 3, respectively.

In order to decompose the input features into two parts: interference features unfavorable to detection and discrimination features favorable to detection, only the output of

the discrimination feature extraction branch is input to the subsequent multi-scale detection module, and the detection loss is used to constrain the branch to learn the discrimination features favorable to target detection. Meanwhile, the orthogonal loss and reconstruction loss between the two branches are used to ensure the learning pair of the interference feature extraction branch detect adverse interference features.

In order to ensure the difference of the features proposed by the two branches, the orthogonal loss between the two branches is defined as follows:

$$L_{diff} = \frac{1}{N}\sum_{n=1}^{N}\left(f_{n1}^{T} \times f_{n2}\right) \tag{1}$$

Where N represents the number of training samples per batch. f_{n1}^{T} represents the column vector drawn from the identification feature graph obtained from the decomposition of the n-th sample. f_{n2} represents the column vector drawn from the interference feature graph obtained from decomposition of the n-th sample. $(\cdot)^{T}$ represents the transpose of the vector.

In order to ensure the integrity of the features proposed by the two branches, the interference features and discrimination features obtained by the two branches are summed and input to the decoder, and the reconstruction loss between the decoder output reconfiguration and the input network image is defined as follows:

$$L_{\text{recon}} = \frac{1}{N}\sum_{n=1}^{N}\|x_n - \hat{x}_n\|_2^2 \tag{2}$$

Where, $\|\cdot\|_2$ indicates a 2 -norm operation. x_n represents the n-th sample. $\hat{x}_n$ represents the reconfiguration output after the n-th sample is passed through the decoder in the feature decomposition module.

2.3. Loss function

The Dice coefficient is a set similarity measurement function, usually used to calculate the similarity of two samples:

$$S = \frac{2|X \cap Y|}{|X| + |Y|} \tag{3}$$

$|X \cap Y|$ represents the intersection between X and Y.$|X|$ and $|Y|$ represent the number of elements of X and Y, respectively. In the study of remote sensing building detection, Dice loss function is applied to binary classification scenario to minimize the similarity between the two categories. Therefore, the Dice loss coefficient can be obtained by inverting the Dice coefficient.

In order to adapt to the training and prediction of binary classification application tasks, this model adopts the loss function combining Dice coefficient and cross entropy loss, and applies it to the output of each layer to carry out backpropagation operation, and finally evaluates the accuracy of the model prediction results.

$$L(Y, \hat{Y}) = -\frac{1}{N}\sum_{b=1}^{N}\left(0.5Y_b \ln \hat{Y}_b + \frac{2Y_b\hat{Y}_b}{Y_b + \hat{Y}_b}\right) \tag{4}$$

3. Experiments and analysis

3.1. Model implementation and training

In this paper, python3.7 is used and the proposed method is verified under the Pytorch framework, where all experimental results are obtained based on Linux server training with an Nvidia GTX 3090 (12G) chip. Each training is 100 rounds, the optimizer is the adaptive learning rate Adam function, and the network backbone freezing method is used to improve the training process and reduce the training time. During model training, the first 50 rounds of the network are frozen. The initial learning rate of the first 50 rounds is $1e^{-4}$, the batch size is 4, and the attenuation coefficient is 0.94. After 50 rounds of unfreezing the network, the initial learning rate of the last 50 rounds is $1e^{-5}$, the batch size is 2 , and the attenuation is once every other round, the attenuation coefficient is 0.94.

3.2. Model Analysis

To evaluate the proposed model, the performance of semantic segmentation is evaluated using recall rate (R), Precision (P), overall average accuracy (OA), goal consistency index (Kappa), and interaction ratio (IoU). The calculation formula is as follows:

$$P = TP/(TP + FP) \tag{5}$$

$$R = TP/(TP + FN) \tag{6}$$

$$F1 = 2PR/(P + R) \tag{7}$$

$$IoU = TP/(TP + FP + FN) \tag{8}$$

$$Kappa = (po - pe)(1 - pe) \tag{9}$$

$$OA = (TP + PN)/(TP + FP + TN + FN) \tag{10}$$

Where P stands for accuracy, R stands for recall rate. TP represents the number of correctly predicted positive samples. FP represents the number of incorrectly predicted positive samples. TN represents the number of correctly predicted negative samples. FN represents the number of incorrectly predicted negative samples. po is the sum of the number of correctly classified samples of each class divided by the total number of samples, that is, the overall classification accuracy. pe is the sum of the product of the actual and predicted quantities corresponding to all

categories, divided by the square of the total number of samples.

In this paper, interaction ratio, accuracy rate, recall rate and Kappa coefficient are used as the main evaluation indicators. Interaction ratio (IoU) represents the ratio between the intersection of building pixels and real positive pixels detected by the model and their union. Precision represents the percentage of real positive pixels in the building pixels detected by the model. Recall represents the percentage of positive pixels of the building detected by the model over the actual positive pixels on the ground. Kappa coefficient is a comprehensive evaluation index of classification accuracy, which is usually used in the evaluation of remote sensing images.

3.3. Data set

In order to verify the network performance proposed in this paper, a high-resolution public remote sensing building image dataset WHU is selected for the experiment [22]. The remote sensing image data is from the New Zealand Land Information Service website, and the ground resolution down-sampled is 0.3m. There are about 22,000 independent buildings in the data set, with weak edge features, complex background information, difficult to distinguish and other characteristics, which has the universality of remote sensing images. Therefore, the experimental results of the set are representative.

This paper cuts the image into 8200, 512×512 pixels, and divides the sample into three parts: the training set (4739), the test set (1035) and the test set (2426). The example is shown in Fig. 2. Before the data training, the data is enhanced by pretreatment, the main purpose is to expand the sample size of the data, the larger the sample size of the data set, the better the results of the model, and the higher generalization ability and robustness performance. In this experiment, the data enhancement is carried out by turning, rotating and tailoring.

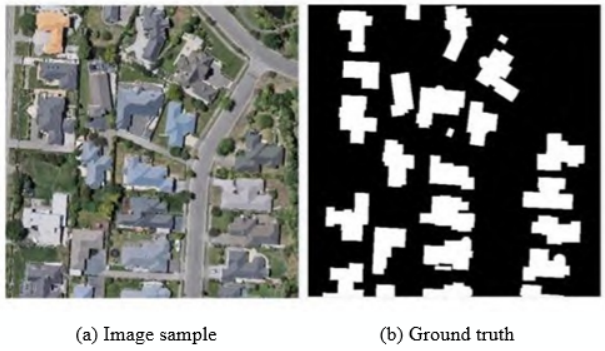

Fig. 2. WHU Dataset Sample

3.4. Experimental results and analysis

In order to verify the validity of the proposed model, the experimental comparison and evaluation with three popular network models (SwinUnet [23], MC-FCU [24], UNet++ [25]) are carried out. In the evaluation results, IoU and Kappa coefficients are mainly concerned to analyze the performance and accuracy of the model, and the results are shown in Table 2.

According to Table 2, the overall accuracy of SwinUnet model is 96.2%, which is 1.3% lower than that of UNet++, indicating that SwinTransformer structure has poor performance in remote sensing building feature extraction tasks and is not suitable for high-resolution remote sensing image detection tasks. The OA value of MC-FCU is 98.3%, which is also lower than UNet++, Proposed model. The results of subjective evaluation are shown in Fig. 3.

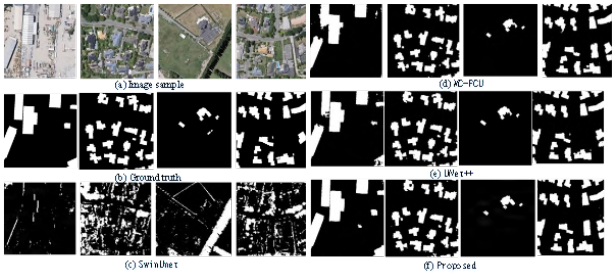

Fig. 3. Prediction results with different models

Meanwhile, parameters of the proposed method were compared with traditional detection networks and the latest remote sensing detection models, as shown in Table 3. The number of parameters in the proposed method is 15.5M, 5.89M more than that of UNet++ and 10.71M more than that of Res_ASPP_UNet++. However, compared with other traditional detection models (such as DeepLabv3, Segnet and FCN [26]), the number of parameters in the proposed method is still small. In the case of a small increase in parameters, the proposed model achieved an IoU coefficient of up to 94.2% and an accuracy of 96.6%, proving that its performance in remote sensing building extraction task is far better than other models, and it retains more feature information for buildings in different locations, and the learned features have higher discriminant ability. Therefore, the model in this paper is more suitable for feature extraction of high-resolution remote sensing images, and fundamentally solves the problem of building edge blur, which has higher application value and practicability.

Finally, through the comparative analysis of each evaluation index, it is concluded that the method in this paper defines the first concurrent network structure that integrates remote sensing image features in an interactive way.

Table 2. Precision result on WHU data set

Model	P/%	R/%	F1/%	Kappa/%	IoU/%	OA/%
SwinUnet	86.9	92.4	89.6	87.3	81.1	96.2
MC-FCU	93.8	96.1	94.9	93.8	90.4	98.3
UNet++	93.7	95.0	94.3	93.1	88.3	97.5
Proposed	96.6	98.4	97.5	96.8	94.2	98.9

Table 3. Model parameter comparison

Model	Parameter/M
Unet	7.85
Unet++	9.61
FCN	18.84
DeepLabv3+	60.12
SegNet	29.54
Double-Unet++	15.1
Unet++-attention	9.81
Res-ASPP-Unet++	4.71
Proposed	15.5

This structure not only naturally inherits the structural advantages of UNet++ and Transformer, but also retains the representation ability of local features and global representations to the maximum extent. The object detection model represented by UNet++ is verified. After the improvement of two-branch network fusion, the problems of feature information loss, fuzzy target edge and background discrimination are solved well. It can be applied to high-resolution building extraction tasks with large sets of training labeled samples.

In order to fully verify the generalization ability of the proposed method in the field of remote sensing image segmentation, the low-resolution Massachusetts dataset was selected for verification. Compared to the WHU dataset, the Massachusetts dataset has low quality and resolution, and there are many error labels. The experimental results show that compared with the WHU dataset, The performance of the proposed method on the Massachusetts data set is degraded, and its IoU, accuracy and recall rate are reduced by 26%,18.6% and 13.9%, respectively. IoU of the Massachusetts data set can be observed and accuracy/recall rates are 30% and 20% lower, respectively, than the WHU dataset, as shown in Table 4. In order to adapt to the use of GPU memory, this paper adopts the sliding window method to cut the images in the training set into 2192 images with the size of 512×512 pixels for training and prediction.

Compared with Unet and Unet ++, the IoU coefficient and Kappa coefficient of the proposed method on the Massachusetts dataset increased by 4.8% and 4.5% respectively,

Table 4. Comparison between WHU database and Massachusetts database/%

Data	P	R	IoU
WHU	94.8	96.1	90.4
Massachusetts	76.2	82.2	65.0

and the results are shown in Table 5. Experiments show that the proposed method has good generalization ability on data sets with different resolutions, and shows that the proposed method has significantly improved the building extraction effect, achieving higher IoU coefficient and Kappa coefficient. This means that the proposed method can segment buildings more accurately and capture the edge details of buildings at the pixel level, thus improving the accuracy and precision of building extraction. However, due to the low resolution of the data set, it leads to missing and false detection. Through the verification of different data sets, it can be proved that the proposed method can significantly improve the feature extraction effect and effectively solve the edge blur problem on both high resolution and low resolution data sets, providing a reliable and efficient solution for remote sensing image segmentation.

Table 5. Precision analysis on Massachusetts model/%

Model	P	R	F1	Kappa	IoU	OA
Unet	65.6	80.9	72.4	68.6	56.6	94.2
Unet++	76.2	82.2	79.1	75.9	65.0	95.6
Proposed	77.9	87.9	82.6	80.4	69.8	97.1

4. Conclusion

This paper studies a building object detection method based on feature decomposition SSD, and solves the problem of many false alarms and missing alarms in complex scene image detection by existing methods. In this paper, by constructing a feature decomposition module, the input features are decomposed into two parts: discrimination features that are favorable to target detection and interference features that are unfavorable to target detection. Among them, the interference features contain complex

background clutter that is easy to cause false alarms, while the discrimination features contain targets of interest. Finally, only the discrimination features are used for multi-scale target detection. The feature decomposition module proposed in this paper does not depend on specific target detection algorithm, and can be applied to any SSD-based target detector. The experimental results based on the measured data set show that the proposed method can significantly reduce false alarms and improve the performance of building object detection in complex scenes.

References

[1] P. Wang, B. Bayram, and E. Sertel, (2022) *"A comprehensive review on deep learning based remote sensing image super-resolution methods"* **Earth-Science Reviews *232***: 104110.

[2] S. Yin, (2023) *"Object Detection Based on Deep Learning: A Brief Review"* **IJLAI Transactions on Science and Engineering *1***(02): 1–6.

[3] Y. Ye, T. Tang, B. Zhu, C. Yang, B. Li, and S. Hao, (2022) *"A multiscale framework with unsupervised learning for remote sensing image registration"* **IEEE Transactions on Geoscience and Remote Sensing *60***: 1–15.

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik, (2015) *"Region-based convolutional networks for accurate object detection and segmentation"* **IEEE transactions on pattern analysis and machine intelligence *38***(1): 142–158.

[5] R. Girshick. "Fast r-cnn". In: *Proceedings of the IEEE international conference on computer vision*. 2015, 1440–1448.

[6] S. Ren, K. He, R. Girshick, and J. Sun, (2016) *"Faster R-CNN: Towards real-time object detection with region proposal networks"* **IEEE transactions on pattern analysis and machine intelligence *39***(6): 1137–1149.

[7] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. "Ssd: Single shot multibox detector". In: *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer. 2016, 21–37.

[8] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. "You only look once: Unified, real-time object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, 779–788.

[9] B. Liu, J. Luo, and H. Huang, (2020) *"Toward automatic quantification of knee osteoarthritis severity using improved Faster R-CNN"* **International journal of computer assisted radiology and surgery *15***: 457–466.

[10] L. Shen, H. Tao, Y. Ni, Y. Wang, and V. Stojanovic, (2023) *"Improved YOLOv3 model with feature map cropping for multi-scale road object detection"* **Measurement Science and Technology *34***(4): 045406.

[11] J. Jang, D. Van, H. Jang, D. H. Baik, S. Duk Yoo, J. Park, S. Mhin, J. Mazumder, and S. H. Lee, (2020) *"Residual neural network-based fully convolutional network for microstructure segmentation"* **Science and Technology of Welding and Joining *25***(4): 282–289.

[12] Y. Liu, B. Fan, L. Wang, J. Bai, S. Xiang, and C. Pan, (2018) *"Semantic labeling in very high resolution images via a self-cascaded convolutional neural network"* **ISPRS journal of photogrammetry and remote sensing *145***: 78–95.

[13] J. Ding, J. Zhang, Z. Zhan, X. Tang, and X. Wang, (2022) *"A precision efficient method for collapsed building detection in post-earthquake UAV images based on the improved NMS algorithm and Faster R-CNN"* **Remote Sensing *14***(3): 663.

[14] T. Bai, Y. Pang, J. Wang, K. Han, J. Luo, H. Wang, J. Lin, J. Wu, and H. Zhang, (2020) *"An optimized faster R-CNN method based on DRNet and RoI align for building detection in remote sensing images"* **Remote Sensing *12***(5): 762.

[15] G. Hua, M. Liao, S. Tian, Y. Zhang, and W. Zou, (2023) *"Multiple relational learning network for joint referring expression comprehension and segmentation"* **IEEE Transactions on Multimedia**:

[16] W. Gaihua, L. Jinheng, C. Lei, D. Yingying, and Z. Tianlun, (2022) *"Instance segmentation convolutional neural network based on multi-scale attention mechanism"* **Plos one *17***(1): e0263134.

[17] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, (2016) *"SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size"* **arXiv preprint arXiv:1602.07360**:

[18] D. Sinha and M. El-Sharkawy. "Thin mobilenet: An enhanced mobilenet architecture". In: *2019 IEEE 10th annual ubiquitous computing, electronics & mobile communication conference (UEMCON)*. IEEE. 2019, 0280–0285.

[19] M. Tan, R. Pang, and Q. V. Le. "Efficientdet: Scalable and efficient object detection". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, 10781–10790.

[20] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu. "Ghostnet: More features from cheap operations". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, 1580–1589.

[21] S. Yin, L. Wang, Q. Wang, M. Ivanović, and J. Yang, (2023) *"M2F2-RCNN: Multi-functional faster RCNN based on multi-scale feature fusion for region search in remote sensing images"* **Computer Science and Information Systems** (00): 54–54.

[22] M. Luo, S. Ji, and S. Wei, (2023) *"A diverse large-scale building dataset and a novel plug-and-play domain generalization method for building extraction"* **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing**:

[23] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, and M. Wang. "Swin-unet: Unet-like pure transformer for medical image segmentation". In: *European conference on computer vision*. Springer. 2022, 205–218.

[24] G. Wu, X. Shao, Z. Guo, Q. Chen, W. Yuan, X. Shi, Y. Xu, and R. Shibasaki, (2018) *"Automatic building segmentation of aerial imagery using multi-constraint fully convolutional networks"* **Remote Sensing *10***(3): 407.

[25] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, (2019) *"Unet++: Redesigning skip connections to exploit multiscale features in image segmentation"* **IEEE transactions on medical imaging *39***(6): 1856–1867.

[26] L. Teng, Y. Qiao, M. Shafiq, G. Srivastava, A. R. Javed, T. R. Gadekallu, and S. Yin, (2023) *"FLPK-BiSeNet: Federated learning based on priori knowledge and bilateral segmentation network for image edge extraction"* **IEEE Transactions on Network and Service Management**: