

Intelligent monitoring of indoor surveillance video based on deep learning

Yun-Xia Liu***, Yang Yang***, Aijun Shi***, Peng Jigang****, Liu Haowei****

*School of Information Science and Engineering, University of Jinan, Jinan 250022, China

**Shandong Provincial Key Laboratory of Network Based Intelligent Computing, University of Jinan, Jinan 250022, China

***School of Information Science and Engineering, Shandong University, Qingdao, Shandong Province, China

****Shandong province's electronic information products quality supervision and inspection Institute, Jinan, China

ise_liuyx@ujn.edu.cn***, yyang@sdu.edu.cn***

Abstract—With the rapid development of information technology, video surveillance system has become a key part in the security and protection system of modern cities. Especially in prisons, surveillance cameras could be found almost everywhere. However, with the continuous expansion of the surveillance network, surveillance cameras not only bring convenience, but also produce a massive amount of monitoring data, which poses huge challenges to storage, analytics and retrieval. The smart monitoring system equipped with intelligent video analytics technology can monitor as well as pre-alarm abnormal events or behaviours, which is a hot research direction in the field of surveillance. This paper combines deep learning methods, using the state-of-the-art framework for instance segmentation, called Mask R-CNN, to train the fine-tuning network on our datasets, which can efficiently detect objects in a video image while simultaneously generating a high-quality segmentation mask for each instance. The experiment show that our network is simple to train and easy to generalize to other datasets, and the mask average precision is nearly up to 98.5% on our own datasets.

Keywords—Surveillance Video; Deep Learning; Mask R-CNN; Object Detection; Instance Segmentation.

I. INTRODUCTION

Traditional video surveillance can only provide simple functions such as video capture and storage. It cannot automatically alarm for abnormal situations. In order to find abnormal behaviours in monitoring in real time, monitor personnel need to constantly observe the video. In this case, the monitor faces dozens of surveillance video images, which is easy to fatigue. It may not be able to respond in time to abnormal situations due to lack of concentration, and loss key information in the video. In addition, because a large amount of surveillance video needs to be stored for months or years, it will result in a large storage cost. Therefore, the intelligent video surveillance system is urgently needed to assist the monitoring personnel to use the intelligent detection technology to process, analyse and understand the video signal while retaining the original video key information, and automatically detect the target category and location information without manual intervention. In the event of an

abnormality, an alarm is issued in time to effectively assist the monitoring personnel.

The traditional moving target detection method can only detect frames with moving targets, but cannot understand the semantic content in the video. With the development of deep learning, advanced target detection, semantic understanding, and instance segmentation techniques are emerging, which realizes the semantic understanding of video content and improves the accuracy. K. He et al. [1] proposed an example-level method for target segmentation, which generates a bounding box and mask for each instance object in the image. The paper was selected as the best paper of ICCV2017.

In recent years, with the advancement of video analysis technology, intelligent video surveillance systems have developed rapidly, and a large number of representative technical achievements and applications have emerged. With the help of embedded video analysis algorithms, the intelligent surveillance camera can identify the abnormal behaviour of the monitoring site without manual intervention. Typical anomalous behaviours include entering and leaving an area, moving quickly, gathering, and so on. In [2], J. Zong et al. proposed a system for motion detection and target recognition, which uses the difference between adjacent frames to mark moving objects and self-mapping neural networks to improve the accuracy of detection. IBM's Intelligent Surveillance System [3] uses the latest computer vision algorithms for automatic event detection in urban surveillance scenarios, specifically for handling crowded environments. X.Liu et al. implemented semantic segmentation based video segmentation and object tracking in [4][5], which facilitates video data analysis.

In this paper, based on reference [1], the key frames containing useful information in the video are extracted to create the required data set. Using the simple, flexible and generic instance object segmentation framework proposed in [1], the Mask R-CNN weight model pre-trained on the MS COCO dataset [6] is used on the dataset of this paper, on the Mask R-CNN. The upper-layer network performs fine-tuning training to obtain the model with the highest accuracy rate adapted to the data set of this paper. Target detection and instance segmentation are performed for videos of 6

consecutive working days in a week, and the prisoners and chairs at various moments in the video are output in the form of text. The number and their location information, plot the daily curve of the number of people over time, and the distribution of the position of the person and the chair, so that the monitor can analyse the difference between the curve under normal and abnormal conditions, and automatically alarm when the abnormal situation is realized.

II. KEY TECHNOLOGIES FOR TARGET DETECTION

Convolutional Neural Networks (CNN) is an artificial neural network, first proposed by LeCun et al., as one of the deep learning algorithms, it has been successfully applied to computer vision. It is currently in the field of image recognition and speech analysis. Research hotspots. It has the weight sharing property, has no pressure on high-dimensional data processing, can make the image directly as the input of the network, and has strong generalization ability. The VGG16 network [8] is taken as an example to analyse the network structure, the BP algorithm used in the training process, and the training process.

Convolutional neural networks have many common classical structures. They started in LeNet in 1998 and became AlexNet in 2012. Since then, they have been popular in image related fields, including ZF-Net, GoogleNet, VGG-Net and ResNet. Below is VGG16. The network is an example to introduce the composition of CNN. The network structure is shown in Figure 1.

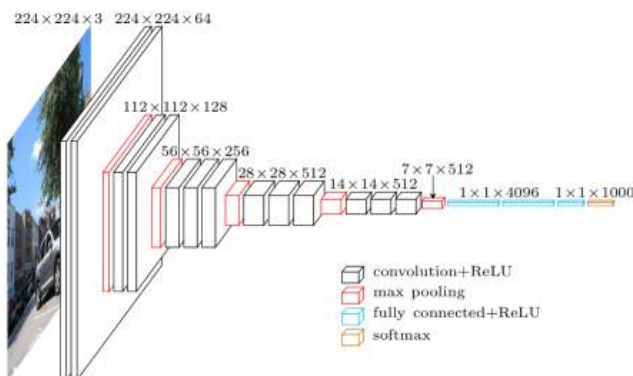


Figure 1. VGG16 network structure diagram

The region-based convolutional neural network method (The Region-based CNN, R-CNN) [10] generates candidate target regions through the selective search method [11], allowing each region of interest (RoI) to pass alone. The convolutional neural network uses the SVM classifier to determine the target category, and uses the border regression to correct the position of the border to achieve the detection of the bounding box target.

Keras is a framework developed by Google's engineer Francois Chollet and a deep learning modeling environment in Python with Tensorflow, CNTK or Theano as the actual computing backend [9]. Compared to several commonly used deep learning frameworks, Keras has a huge advantage in that it is easy to use, frees users from complex mathematical

formula commands, and directly considers the architecture of specific deep learning neural networks.

Keras has several significant advantages in practical applications [9]:

(1) Keras provides some highly efficient Python APIs that can be run on other deep learning frameworks, supporting existing common structures such as CNN and RNN, enabling users to quickly build and train their own deep learning models.

(2) Lightweight and highly modular, users can arbitrarily combine modules to construct the required model. In Keras, any neural network model can be described by a sequence model or a graph model. The components are divided into the following modules: neural network layer, activation function, loss function, regularization method, initialization method, and optimization engine. Users can put these modules into a sequence model or graph model in a reasonable way to construct the required model, reducing the amount of code writing work, high efficiency and not easy to make mistakes.

(3) Based on Python, ease of use and scalability are very high.

(4) It can seamlessly switch between CPU and GPU, suitable for different application environments.

III. KEY TECHNOLOGIES FOR MASK R-CNN

The network structure is shown in Figure 2. Generally, the Mask R-CNN is divided into two parts: the lower layer backbone network and the upper layer network. The lower layer network is the ResNet-FPN convolutional neural network. The main function is to extract the characteristics of the input image. The upper layer network is in the Faster R-CNN adds an FCN based on classification, border regression and mask prediction.

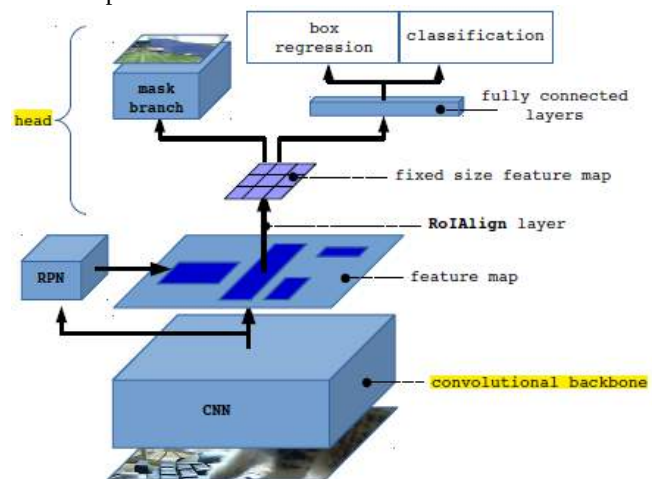


Figure 2. Mask R-CNN network structure diagram

The underlying backbone network is represented by the naming manner of the network-depth-feature extraction layer. K.He et al. evaluated ResNet networks [19] and ResNeXt networks [20] with depths of 50 and 101 layers, such as ResNet-50's backbone network using ResNet-50-C4, where

C4 refers to Faster R-CNN from ResNets. The last layer of the fourth convolution phase extracts features.

K. He et al. also adopted another more efficient backbone network proposed by Lin et al., called Feature Pyramid Networks (FPN) [21]. FPN combines low-level semantic information, high-resolution low-level features, high-semantic information, and low-resolution high-level features from top to bottom. The RoI features with rich semantic information can be extracted at different scales.

The upper layer network adds a mask branch to the Faster R-CNN. As shown in Figure 3, the upper gray portion indicates the R-CNN combined with the ResNet or FPN network, and the lower black portion indicates the added FCN for the prediction mask.

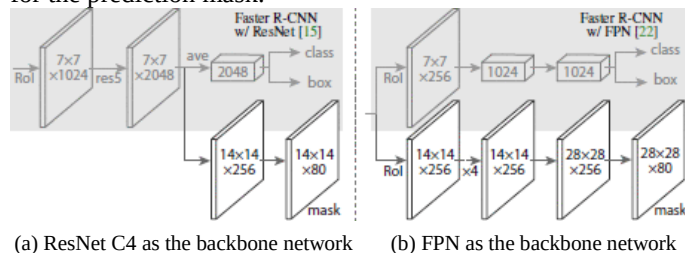


Figure 3. Mask R-CNN upper layer network

Mask R-CNN performs all experiments using the COCO dataset, training on a training set with 80k images and a subset of validation sets (trainval35k) with 35k images, with the remaining subset of validation sets containing 5k images. Control experiments were performed on (minival) and evaluated using the standard COCO indicator AP (average precision). The AP in this paper specifically refers to the mask IoU, including (the accuracy rate when the IoU threshold is 0.5), (when the IoU threshold is 0.75) Accuracy).

(1) Training

Take the picture as the centre for training, first adjust the size of the picture, each GPU processes 2 pictures at a time. After the RPN network, each picture has N samples of RoI, the ratio of positive and negative samples is 1:3, C4 backbone The N is 64 and the FPN is 512. The learning rate is 0.02, and 160k iterations are trained on 8 GPUs (the size of the effective small batch is 16). With the idea of learning rate attenuation, the learning rate is reduced by 10 times in the 120th iteration. The weight attenuation is 0.0001 and the momentum is 0.9. The RPN anchor spans 5 scales and 3 aspect ratios.

(2) Forecast

In the prediction, the number of candidate regions of the C4 backbone network is 300, and the number of FPNs is 1000. Perform bounding box prediction on these candidate regions, perform non-maximum suppression (NMS) [22], and then apply the mask branch to the top 100 detection boxes with higher scores, for each RoI predicts K masks, but only selects the mask of the k^{th} category of the classification branch prediction.

IV. VIDEO INTELLIGENT DETECTION BASED ON MASK R-CNN METHOD

Compared with Faster R-CNN, Mask R-CNN only adds a small overhead, training is simple and versatile. This paper

uses Mask R-CNN method for indoor surveillance video intelligent detection. Because the pre-trained weight model on the COCO dataset can identify 80 types of targets and backgrounds, and the types of recognition are large, the AP values on the datasets in this paper are not high, and the results are shown in Table 6. In this paper, we only need to identify the two types of targets and background, the number of statistical targets and the location distribution. Therefore, we use the migration learning idea to fine-tune the Mask R-CNN network, and use the pre-trained COCO weight model in this paper. Retrain the upper layer network to get the best model to detect the number and position information of people and chairs at various moments in the video, draw the number and distribution curve, and realize automatic alarm of abnormal events.

This article uses a video recorded by a surveillance camera in a school lab to create a data set. A video of Monday to Saturday workdays was recorded according to the life schedule of 6 working days in a week. First extract the key frames in the video, then mark the outline and category of the target in the image, save the marked image as a json file, and make a standard data format that Mask R-CNN can process. The specific process is as follows. The lab members first recorded the 6-day video according to the routine of life. The frame rate of the surveillance video was 25 frames per second (fps), and the gap between adjacent frames was small. If each frame was extracted, there would be a large number of Similar pictures, so choose to take one frame every 12 seconds, that is, take one frame every 288 frames, thus converting one day's video into about 7,500 pictures.

However, there are still a large number of duplicate pictures in the preliminary extracted pictures. For example, when sleeping in a dormitory or going out to work, this has little effect on training the neural network, so we extract the frames in the picture sequence whose target changes greatly. Through the above two steps, more than 600 useful images were extracted from the original large and redundant data set.

After that, under the ubuntu16.04 system, use the labelme annotation software to mark the picture, draw the mask of the person and the chair in the picture, and mark the category. The naming rules are person1, person2...chair1, chair2..., as shown in Figure 4. Save as a json file. Then use labelme_json_to_dataset to convert the json file into the required label folder. There are five files in each folder, namely img.png, info.yaml, label.png, label_names.txt and label_viz.png, which need to be used. Only info.yaml and label.png are available.



Figure 4. The process of labeling images with labelme

However, the mask label label.png generated by labelme is stored in 16 bits, and the default reading mode of opencv is 8 bits, so the 16-bit mask label is further converted into an 8-bit mask label. At this point, the data set is completed, and the data set of this article is named Lab426. Finally, the labeled data set is divided into a training set containing 300 pictures and a verification set containing 40 pictures, and the remaining 200 pictures not marked are used as test sets.



Figure 5. Label map during dataset production

V. EXPERIMENTAL RESULTS

Unlike the learning rate in [1], Matterport's open source code on github [23] uses a learning rate of 0.001. Because the data set in this paper is relatively small, and ResNet-101-FPN is used as the lower convolutional backbone network in [1], the pre-trained COCO weight model obtained by training on the 2014 MS COCO data set has been well trained, so the migration is utilized. The idea of deep learning, using the weight model as a starting point to train a new model on its own data set, you can not modify too many weights in a short time, only fine-tuning.

There are two ways for pre-training model:

(1) Training only the upper layer network part: generally applicable to the new data set is similar to the original data set, and the new data set is small. In order not to destroy the feature extraction capability of the underlying backbone network, all ResNet-101-FPN layers are frozen, and the weight values of these lower-layer backbone networks are kept unchanged, and only the randomly initialized layer, that is, the layer of the upper layer network is trained.

(2) Training all layers: In order to better adapt the new data set, all layers need to be retrained, that is, the weight of the pre-training is used as the initial value, from the first layer of the network to the last layer.

Because the dataset of this paper is similar to the MS COCO dataset, and the dataset of this paper is relatively small, in order to ensure the fastest convergence and save time under the premise of ensuring the accuracy, we use the first method to train on one GPU (effective The small batch size is 2) and 3k iterations are performed, and the learning rate is 0.001.

1) Training process

During the training process, a picture is randomly selected from the training set, and the mask of different objects in the two types of objects of the person and the chair is visualized as shown in Figure. 6.

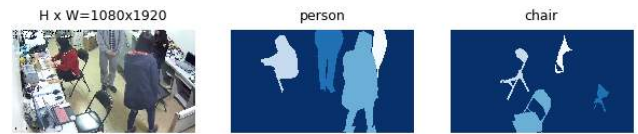


Figure 6. Target mask visualization during training

During the training process, the fit_generate() function automatically saves the training log to the logs file. In this paper, the tensorboard tool in the TensorFlow framework is used to observe the change of each loss function in real time. According to the change trend of the loss function with the number of iterations, the network parameters are continuously adjusted. When the loss function is declining and tends to converge, the network model has been optimized. In the training set with 300 pictures and the verification set containing 40 pictures, the network parameters are set to Table 4, the variation curves of the loss function of the training set and the verification set with the number of iterations are shown in Figure. 7 and Figure. 8.

TABLE 1. MASK R-CNN NETWORK PARAMETERS

(a) Definition of positive and negative samples					
samples	Definition				(%)
Positive	IoU of RoI and ground-truth box ≥ 0.5				25
Negative	IoU of RoI and ground-truth box < 0.5				75

(b) Training parameter					
Learning rate	weight attenuation	batch size	epoch	steps per epoch	validation steps
0.001	0.0001	2	19	150	20

(c) RPN parameter				
Positive sample ratio	anchor ratios	anchor scales	anchors per image	NMS Threshold
0.33	0.5, 1, 2	32, 64, 128, 256, 512	256	0.7

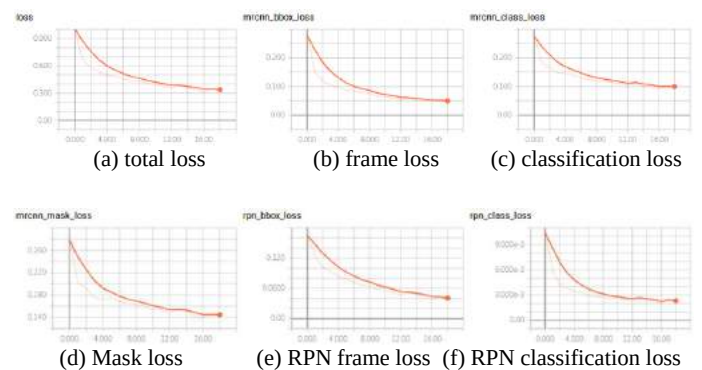


Figure 7. Training set loss function curve (light line is the original curve, dark line is the smooth curve)

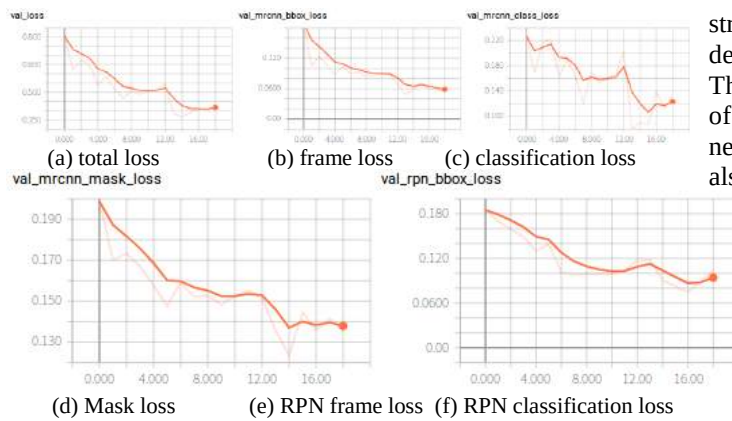


Figure 8. validation set loss function curve (light line is the original curve, dark line is the smooth curve)

When the training set size is 300, this paper compares the impact of the anchor size of the RPN network on the network accuracy. As shown in Table 2, when the aspect ratio of the anchor is fixed at (0.5, 1, 2), the value of the mAP is the highest when the anchor is (32, 64, 128, 256, 512) because the original image The size of these five scales is the best match with the size of each target in the picture. It will not detect the larger target because the anchor is too small, and it will not detect the small target because the anchor is too large. .

TABLE 2. ANCHOR SIZE

anchor scales	mAP ₅₀ (%)	mAP ₇₅ (%)
[16, 32, 64, 128, 256]	94.2	80.7
[32, 64, 128, 256, 512]	98.5	93.9
[64, 128, 256, 512, 1024]	95.5	89.3

When the training set size is 100, the effects of different size batches on network training speed and model accuracy are compared. This article uses 1 GPU, steps=100/batch_size in an epoch. As shown in Table 3, the total training time is almost equal when the batch is 1 and 2. The training time of a picture is about 10s, and the loss is 1 when the batch is 1. The function oscillates and converges. When the batch is 2, the number of iterations required for one cycle is reduced. When the size of the batch is 4, it cannot run because of insufficient memory.

TABLE 3. BATCH SIZE

batch_size	run_speed(s/step)	steps	mAP ₅₀ (%)	mAP ₇₅ (%)
1	10	100	96.5	86.1
2	20	50	96.6	86.6

This article originally used a training set of 100 images and a validation set of 40 images for training. Considering that the training set is too small, there is over-fitting, so the training set is increased to 200 and 300, as shown in Table 4. The impact of training set size on network accuracy. When the network parameters are the same and the epoch is 18, the value of mAP increases with the increase of the training set. The larger the training set, the more extracted features, the

stronger the generalization ability of the network, and the detection and segmentation. The accuracy rate is also higher. Therefore, the follow-up can continue to increase the number of training sets, improve the generalization ability of the network, and reduce over-fitting, but the training time will also increase, increasing the time cost.

TABLE 4. DATA SET SIZE

Lab426	run time(h)	mAP ₅₀ (%)	mAP ₇₅ (%)
100	4.9	95.5	84.8
200	9.5	96.6	86.1
300	14.2	98.5	93.9

Among the top 20 epochs, the Mask R-CNN model with the highest accuracy rate on the verification set was selected as the model. The average accuracy of the COCO weight model and the model on the Lab426 verification set is shown in Table 4. Using the COCO weight model as the initial value, the upper layer network of Mask R-CNN is retrained on the data set of this paper. Through continuous adjustment, the mask AP of the final model is increased to 98.5%. The test results of the two models on the same picture are shown in Figure 9.

TABLE 5. ACCURACY OF COCO AND LAB426 MODELS

mAP	mAP ₅₀ (%)	mAP ₇₅ (%)
COCO	32.9	29.5
Lab426	98.5	93.9



(a) COCO model test results (b) Lab426 model test results

Figure 9. Test results of COCO and the Lab426 model on the same image

The result of the fine-tuned Mask R-CNN is shown in Figure 10. Good results have been achieved in both target detection and instance segmentation. Even if there is partial occlusion overlap between the targets, better segmentation results can be obtained. However, in the 40 pictures of the verification set, there are still pictures with detection errors, and some pictures are shown in Figure 4.10. As can be seen from the figure, when there is a large overlap between different targets, or due to occlusion problems (as shown in Figure 4.10(a), a person sitting in a chair obscuring the middle part of the chair can only see the chair. The backrest and part of the chair leg, Mask R-CNN will recognize it as two chairs), there will be errors. Since the data set is too small, it is inevitable that the fitting phenomenon occurs. Therefore, in the case of optimizing the training speed, the number of training sets can be increased, and the network generalization ability can be enhanced.



Figure 10. Fine-tuned Mask R-CNN test results



Figure 11. Error result is in the test results

CONCLUSIONS

This paper proposes a method based on Mask R-CNN for intelligent monitoring of indoor surveillance video. It uses advanced target detection and instance segmentation technology in deep learning to realize the semantic understanding of video content, which not only effectively retains the key information in the original video. It also reduces the storage and processing pressure of massive monitoring video, and the automatic alarm function for abnormal events reduces the workload of monitoring staff.

REFERENCES

- [2] He K, Gkioxari G, Dollár P, et al. Mask R-CNN[C]// IEEE International Conference on Computer Vision. IEEE, 2017:2980-2988.
- [3] Meng Z J, Jie C. Design on Intelligent Video Surveillance System Based on Target Identification Algorithm[J]. Applied Mechanics & Materials, 2015, 713-715:460-465.
- [4] Tian Y L, Brown L, Hampapur A, et al. IBM smart surveillance system (S3): event based video surveillance system with an open and extensible framework[J]. Machine Vision & Applications, 2008, 19(5-6):315-327.
- [5] Liu X, Song M, Tao D, et al. Random geometric prior forest for multiclass object segmentation[J]. IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society, 2015, 24(10):3060-70.
- [6] Liu X, Tao D, Song M, et al. Weakly Supervised Multiclass Video Segmentation[C]// Computer Vision and Pattern Recognition. IEEE, 2014:57-64.
- [7] Lin T Y, Maire M, Belongie S, et al. Microsoft COCO: Common Objects in Context[J]. 2014, 8693:740-755.
- [8] Michael A. Nielsen, "Neural Networks and Deep Learning", Determination Press, 2015.
- [9] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.
- [10] Mcleod C . A Framework for Distributed Deep Learning Layer Design in Python[J]. Computer Science, 2015, 10(1).

- [11] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, 2014.
- [12] Uijlings J R R, Sande K E A V D, Gevers T, et al. Selective Search for Object Recognition[J]. International Journal of Computer Vision, 2013, 104(2):154-171.
- [13] He K, Zhang X, Ren S, et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2015, 37(9):1904-1916.
- [14] Girshick R. Fast R-CNN[C]// IEEE International Conference on Computer Vision. IEEE Computer Society, 2015:1440-1448.
- [15] Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[C]// International Conference on Neural Information Processing Systems. MIT Press, 2015:91-99.
- [16] Shelhamer E, Long J, Darrell T. Fully Convolutional Networks for Semantic Segmentation[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2014, 39(4):640-651.
- [17] Jaderberg M, Simonyan K, Zisserman A, et al. Spatial Transformer Networks[J]. 2015:2017-2025.
- [18] Dai J, He K, Sun J. Instance-Aware Semantic Segmentation via Multi-task Network Cascades[C]// Computer Vision and Pattern Recognition. IEEE, 2016:3150-3158.
- [19] Pinheiro P O, Collobert R, Doll, et al. Learning to segment object candidates[J]. 2015:1990-1998.
- [20] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[J]. 2015:770-778.
- [21] Xie S, Girshick R, Dollár P, et al. Aggregated Residual Transformations for Deep Neural Networks[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2017:5987-5995.
- [22] Lin T Y, Dollár P, Girshick R, et al. Feature Pyramid Networks for Object Detection[J]. 2016:936-944.
- [23] Girshick R, Iandola F, Darrell T, et al. Deformable part models are convolutional neural networks[J]. 2014:437-446.
- [24] https://github.com/matterport/Mask_RCNN
- [25] Shao Z, Cai J, Wang Z. Smart Monitoring Cameras Driven Intelligent Processing to Big Surveillance Video Data[J]. IEEE Transactions on Big Data, 2018, PP(99):1-1.



Yun-Xia Liu is with the school of information science and engineering, university of Jinan, Shandong Province, China. She achieved the Ph.D. in 2012 in Shandong University and became an associate professor in University of Jinan in 2015. Her research interest includes multi-scale geometry analysis, wavelet analysis, and pattern recognition.



Yang Yang is with the school of information science and engineering, Shandong University, Shandong Province, China. He achieved the Ph.D. in 2009 in Shandong University. His research interest includes image analysis, wavelet analysis, video surveillance and pattern recognition.