Research on the state detection of the secondary panel of the switchgear based on the YOLOv5 network model

Wang Chen¹, Zhang Liqiang¹, Yan Tianpeng¹, Jiang Tao¹, Jia Yijing², Lai Zhihao²

¹Commissioning center, China Nuclear Power Engineering Company Ltd, Guangzhou Shenzhen, 518000, China
²Department of Electrical Engineering, North China Electric Power University, Hebei Baoding, 071000, China

*E-mail: wang_chen@cgnpc.com.cn

Abstract. In recent years, with the development of artificial intelligence algorithms, deep learning algorithms are widely used in target detection. The switchgear in the power system plays a key role in the safe operation of the system and there are a large number of them. In order to improve the work efficiency of testers and reduce manual misjudgment, it is proposed to apply the deep learning YOLOv5 algorithm to detect the status of the switchgear secondary panel in real time. The algorithm uses the configuration environment, the data set training and target test obtain the light state information, and the experimental results prove the effectiveness of the algorithm, which provides technical support for the field of switchgear electrical test robots.

1. Introduction

The main function of the switchgear in the power system is electric energy transmission, and it is the primary equipment with the largest number of operations in the power grid system, so the safe and reliable operation of the switchgear is very important[1]. The status of the secondary panel of the switchgear can reflect the working status of the power distribution device itself, and it is also a necessary inspection step before the electrical test. The detection content of the state of the secondary panel of the switch cabinet is mainly to detect the light on and off and position information, and logically judge whether the state is correct.

The traditional detection method[2] method has long calculation time, window redundancy, and the diversity of the environment such as lighting will affect the robustness of the algorithm. In recent years, deep learning has developed rapidly. The current mainstream deep learning target detection algorithms mainly include two-stage detection algorithms (Two-stage) and single-stage detection algorithms (One-stage). The two-stage detection algorithm is divided into two stages: candidate region and feature extraction, region classification and position adjustment. Typical algorithms include R-CNN[3] and its variant Fast R-CNN[4]; the single-stage detection algorithm directly passes Regression to get the probability and location coordinates of the target, classic algorithms include YOLO[5] (You only look once), SSD[6] (Single Shot MultiBox Ssectector), etc. The former has higher detection accuracy, but its detection speed is slower, while the latter greatly improves the detection speed while maintaining performance[7]. YOLO is a relatively new single-step target detection algorithm that can simultaneously predict the bounding box and category of the entire image in one
scan, without the use of RPN (Region Proposal Network). Compared with the two-step target detection algorithm, it directly generates target location and category information through the network, which has a faster detection speed. YOLOv5 is currently an excellent target detection algorithm. The multi-scale fusion in its network can detect small targets well and has strong real-time performance. It is widely used in various scenarios[8].

In response to the above problems, this paper uses the YOLOv5 algorithm to detect the lighting information of the switchgear secondary panel in real time, trains the coco128 data set and self-made data set provided by the official website, detects the light color and position information in real time, and monitors the operating status of the switchgear. judgment.

2. YOLOv5 detection model

YOLOv5 has 4 models according to the scale of the model. It can set four different weights of v5s, v5m, v5l, v5x, and the depth and width gradually increase. According to the complexity of the research object in this article, it is determined to use YOLOv5s to efficiently complete the experiment. The YOLOv5 network structure is composed of input layer, backbone network, neck and output layer. The structure diagram is shown in Figure 1.

2.1. Configuration Environment

According to the standard of training customized data, the overall process of developing environment configuration is as follows:

1. Create a yaml file to read the configuration file of the training data path for the project;
2. Create the labels information of the data;
3. Organize the project and data file structure;
4. Select the model you want to use in the above and modify the specified parameters;
5. Conduct training and testing.

data is used to place the configuration files of related training data; models is used to place the parameter configuration files of each model; weights are used to place the weight files of the pre-trained model; inference is used to place the test pictures in the prediction/inference phase; runs are placed in the training process. Some data; the files in the current directory focus on train.py, test.py, detect.py, these are the key source code of the network model[9].

Download the weight data of the pre-trained model, that is, the model weight of v5s, and place the downloaded yolov5s.pt file in the current folder. Perform a model test on the pictures in the inference directory. The weight data used is the downloaded yolov5s.pt, and the inferred output pictures are stored in inference/output.
2.2. Training data set

(1) Training the coco128 data set

The coco128 data set refers to the first 128 pictures of the coco data set. It is usually used as a small tutorial data set. The 128 pictures in it are used for training and verification. Place the folder in the same level directory as YOLOv5. coco128.yaml corresponds to the required coco128 data set, open the coco128.yaml file for training. The training method is to use the downloaded pre-trained model weights for training.

YOLOv5 sets the data storage of the training process and the visualization of the training pictures, so that the accuracy of the training data can be known. The training results are sequentially stored in the paths of ./runs/exp0, ./runs/exp1, etc., and subsequent experimental results will also be sequentially stored. The losses and evaluation indicators during training are saved in Tensorboard and ./runs/exp0/results.txt log files, where results.txt will be visualized as results.png at the end of training.

After the training is completed, the best weight best.pt and the last weight last.pt of the whole process will be saved under ./runs/exp*/weights/, and the best weight can be used for single or batch image targets Identify predictions[10].

(2) Training public data set

After training the coco128 data set, you need to train the public data set. Here, the public data set uses the Mask Wearing Dataset, which contains 149 pictures, and selects the marked 416x416-black-padding data for training.

In target detection, the target in the original picture needs to be manually framed, that is, the bounding box, and the target category is set (the category index starts from 0). The position, category and other information of the frame will be saved as the corresponding label file. The label information file in YOLOv5 is a txt file, which contains the center x coordinate x_center, the center y coordinate y_center, width, and height of each target bounding box. Among them, these 4 pieces of information should be normalized to a value between 0-1; and the target category class is added before these 4 pieces of information, and the index of the category needs to start from 0.

The process of training and prediction is the same as above, the instruction only needs to set the --data parameter to --data./data/mask.yaml, and the others are the same as above.

(3) Training self-made data set

As shown in Figure 2, photos of different lights and different angles of the simulated lighting panel were taken to make the data set.
Figure 4. Part of the self-made data set.

When you have the original image, the most important thing is to frame the data. Here, the LabelImg annotation tool is used to annotate the image. LabelImg can directly select the required format label to save the file, which is more convenient and simple[11].

The self-made data set needs to be listed in the same level directory as above, and the total data set folder should be placed in the same level directory of /yolov5. For training, you can also set the --data parameter to --data./data/my_datasets.yaml. Since it is training on a self-made data set, the training algebra should be set as large as possible to be effective.

3. Experimental results and analysis

3.1. The image characteristics of the secondary panel lighting of the power high-voltage switchgear

Usually in the high-voltage switch room of the substation, the size of the switch cabinet is basically the same, and the interval is even and neatly arranged.

Figure 5. Image of high-voltage switchgear. Figure 6. Image of the secondary panel of the switchgear.

3.2. Image preprocessing

According to the characteristics of the light secondary panel, in order to improve the recognition accuracy of the image and improve the image quality, the image must first be preprocessed. The specific process is as follows:

(1) Gaussian smoothing

The purpose of Gaussian smoothing is to remove noise in the image. When the light is relatively weak, it will cause more noise in the captured image. Gaussian smoothing of the image is to do a weighted average of each pixel in the field and determine the weighted value through the Gaussian function.

(2) Enhance image contrast

Because the light in some switch cabinets may be relatively dark, enhancing the image contrast during preprocessing can improve the accuracy of image recognition. By adjusting the contrast of the image, you can adjust the gray value of the area with low gray value to a lower gray value, and adjust the gray value of the area with high gray value to higher, so that the contrast between light and dark of the image is more obvious to highlight the details. Contrast[12].
3.3. YOLOv5 outputs detected target information

Firstly, judge the status of the indicator light by color or position. If the indicator light status before the ground knife action is shown in Figure 4, then this status is the correct status, and the other lights are on or all off are the error status. Therefore, we need to analyze the status according to the pictures obtained by the camera, and get the conclusion of right or wrong.

Observe the status of the indicator light of the switch cabinet through the camera. If it is correct, proceed to the next step. The following are mainly divided into these steps:

(1) Acquire images

First, open the connected camera device by calling the camera's API, and then the camera will transfer the collected real-time image data to the data buffer, and process the image when it reaches the designated collection area.

(2) Judgment of light status

Through the target detection algorithm, the four circular lights on the secondary panel are identified and the coordinate information is obtained. Sort the color information according to the coordinates, and finally judge whether the state is correct or not through the pixel information.

Find plot_one_box from detect.py, enter general.py to automatically locate the plot_one_box function, modify the function, The output target coordinate information is shown in Figure 7.

![Figure 7. Output target coordinate information.](image1)

![Figure 8. Pixel detection area.](image2)

Obtain the pixel at the position of the indicator, and determine the pixel range of on and off according to the pixel value. However, because the light-off state is similar to the color of other unlit lights, false alarms are likely to occur when detecting light-off pixels. The light pixels of the light are more obvious, and the light only has two states, the light is on and the light is off, so only the pixel range of the light needs to be detected. It can be understood as a bright light within the determined pixel range. If it is not within this pixel range, it means the light is off.

After determining the pixel range for lighting, monitor the pixels in the selected area in real time, and determine whether any pixels appear within the specified range[13]. Once the pixel is within the range, it means that the corresponding indicator light is on. Therefore, a threshold value needs to be set, and when this threshold value is reached, the fire alarm light is considered to be on. However, due to different ambient light, once the reflection occurs, it will also cause some points to meet the range, so a large number of pixels must be checked for judgment.

Get the pixels of the fire alarm light position, and select the pixel range of the light on and off according to the pixel value. The detected area is shown in Figure 8.

This is a 22×22 square area, because the illuminated area is a circular area. To ensure accuracy, an image with an image center distance of 11 pixels is detected. If you want higher accuracy, we can narrow the range. For example, if the distance to the image center is 10, the detected results are shown in Table 1.

![Table 1](image3)

(3) When the light status is correct

The number of pixels in the middle of a circle with a radius of 10 must be greater than the number of pixels in the middle of a square with a side length of 10. A square with a side length of 10 has 100 pixels. Then our threshold can be set to 100, as long as we detect that the number of pixels that meet the above range is greater than 100, the fire alarm light is considered to be on.

The indicator light is on and the result is correct, you can proceed to the next step.
Table 1. Pixel detection result.

| Color channel | range    |
|---------------|----------|
| B             | [125,255]|
| G             | [149,255]|
| R             | [223,255]|

4. Conclusion
Aiming at the problem of identifying the secondary panel of the switchgear, a convolutional neural network algorithm under the YOLOv5 framework based on OpenCV pixel detection is proposed. Completed the light detection, and judged the color coordinates and other information, quickly and efficiently concluded that the status of the secondary panel was correct or not, and prepared for the next action of the switchgear electrical test robot. It can also be trained to perform similar operations on other hardware judgment.

References
[1] Qin Yu, Fan W N, Zhang H, et al. (2018) Partial discharge on-line detection and comprehensive diagnosis analysis of insulation defects in switchgear[J]. High Voltage Apparatus, 54(11): 278-283.
[2] Uijlings J R, Van De Sande K E A, Gevers T, et al. (2013) Selective Search for Object Recognition[J]. International Journal of Computer Vision, 104(2):154-171.
[3] Girshick R, Donahue J, Darrell T and Malik J, (2014) "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 580-587, doi: 10.1109/CVPR.2014.81.
[4] Girshick R. Fast R-CNN[J]. (2015) Computer Science.
[5] Redmon J, Divvala S , Girshick R , et al. (2016) You Only Look Once: Unified, Real-Time Object Detection[C]// Computer Vision & Pattern Recognition. IEEE.
[6] Liu W, Anguelov D, Erhan D, et al. (2016) SSD: Single Shot MultiBox Detector[C]// European Conference on Computer Vision. Springer, Cham.
[7] Yan H W, Chen J X. (2020) Insulator String Positioning and State Recognition Method Based on Improved YOLOv3 Algorithm[J]. High Voltage Engineering, 46(2): 423-432.
[8] Zheng H B, Li J H, et al. (2021) Infrared Object Detection Model for Power Equipment Based on Improved YOLOv3[J]. Transactions of China Electrotechnical Society, 36(07):1389-1398.
[9] Ye A, Pang B, Jin Y, et al. (2020) A YOLO-based Neural Network with VAE for Intelligent Garbage Detection and Classification[C]// ACAI 2020 3rd International Conference on Algorithms, Computing and Artificial Intelligence. 2020.
[10] Zheng T T, Yang X, Dai Y. (2020) Overview of Anchor Free Object Detection Model Based on Key Points[J]. Computer Systems & Applications, (8):1-8.
[11] Peng X, Xiao Y, Cui Z, et al. (2019) Research on Image Recognition of Opening and Closing State of Power Switchgear Cabinet Door Based on Convolutional Neural Network[J]. Bulletin of Science and Technology, 35(11):94-98.
[12] Yang Yu M, Liao Yu R, Lin C B, et al. (2021) A Survey of Object Detection Algorithms for Lightweight Convolutional Neural Networks[J]. Ship Electronic Engineering, 41(04):31-36.