A Light-Weight Network for Small Insulator and Defect Detection Using UAV Imaging Based on Improved YOLOv5

Tong Zhang , Yinan Zhang * , Min Xin , Jiashe Liao , Qingfeng Xie

Posted Date: 11 May 2023

doi: 10.20944/preprints202305.0796.v1

Keywords: keyword light-weight; insulator and defect detection; YOLOv5; Ghost module; convolutional block attention module; unmanned aerial vehicles

Article

A Light-Weight Network for Small Insulator and Defect Detection Using UAV Imaging Based on Improved YOLOv5

Tong Zhang [1], Yinan Zhang [2,*], Min Xin [3], Jiashe Liao [4] and Qingfeng Xie [5]

[1] Guilin University of Electronic Technology, Guilin 541004, China; mala@guet.edu.cn
[2] Guilin University of Electronic Technology, Guilin 541004, China; 1205440049@mails.guet.edu.cn
[3] Guilin University of Electronic Technology, Guilin 541004, China; xinmin1203@163.com
[4] Guilin University of Electronic Technology, Guilin 541004, China; 20012201020@mails.guet.edu.cn
[5] Guilin University of Electronic Technology, Guilin 541004, China; 18966409631@163.com
* Correspondence: 1205440049@mails.guet.edu.cn

Abstract: Insulator defect detection is of great significance to compromise the stability of the power transmission line. The state-of-the-art network of object detection, YOLOv5, has been widely used on insulator and defect detection. However, YOLOv5 network has some limitations like poor detection rate and high computational loads in detecting small insulator defects. To solve these problems, we proposed a light-weight network for insulator and defect detection. In this network, we introduced Ghost module into YOLOv5 backbone and neck to reduce the parameters and model size to enhance the performance in unmanned aerial vehicles (UAVs). Besides, we added small object detection anchors and layers for small defect detection. In addition, we optimized the backbone of YOLOv5 by applying convolutional block attention module (CBAM) to focus on critical information for insulator and defect detection and suppress uncritical information. The experiment result shows the mean average precision (mAP) 0.5 and the mAP0.5:0.95 of our model can reach 99.4% and 91.7%, the parameters and model weight are reduced to 3807372 and 8.79M, which can easily deploy to embedded devices like UAVs. And the speed of detection can reach 10.9ms/image, which can meet the real-time detection requirement.

Keywords: light-weight; insulator and defect detection; YOLOv5; Ghost module; convolutional block attention module; unmanned aerial vehicles

1. Introduction

Insulators in power system play an irreplaceable role in mechanical support and electrical insulation of transmission lines. Most insulators in transmission lines operate outdoors and are subject to a variety of environmental interference [1]. In the event of insulators failure or explosion, it can affect power supply and cause great damage to the power transmission system [2].

In recent years, self-explosion and branch-drop of insulators has become a common problem in high-voltage transmission lines and it is very hard to check for insulator defects manually. Therefore, using UAVs has become increasingly common to capture insulator images and identify defective insulators by deploying the object detection algorithm to the embedded devices [3,4].

We can divide object detection methods into two types [5]. One is traditional object detection method, like Support Vector Machine (SVM) [6], Viola Jones (VJ) detector [7], Deformable Parts Mode (DPM) detector [8] and Histogram of Oriented Gradients (HOG) detector [9]. However, traditional object detection algorithm selects features largely depend on manual work, and in dealing with challenges like complex background and large camera dynamics, these methods have poor robustness and low accuracy [10].

Another method of object detection is based on deep learning, which includes one-stage algorithm and two-stage algorithm. One-stage algorithm extracts feature maps through convolution

 doi:10.20944/preprints202305.0796.v1

neural network to infer the classification and bounding boxes of the target. Two-stage algorithm first extracts region proposals, and infers the classification and location of the target through the convolution network.

Girshick et al. first proposed R-CNN [11] in 2014, which is two-stage algorithm, it applied convolution neural network to extract features, and with the good performance of CNN, the accuracy from PASCAL VOC dataset is raised from 35.1% to 53.7%. Then they proposed Fast R-CNN [12] and Faster R-CNN [13] to improve the performance of detection. Wang et al. [14] proposed a method with improved Faster R-CNN combined with the backbone of Resnet 50 for insulator detection. However, its low detection speed and large numbers of parameters cannot meet the real-time detection requirement.

In contrast, Redmon et al. proposed one-stage algorithm You Only Look Once (YOLO) [15] series, which are mainstream object detection algorithms. Compared with R-CNN series, YOLO series are more suitable for applications of real-time detection. Yang et al [16]. proposed improved YOLOv3 insulator detection method. This method added additional max pooling layer and Smooth-EIoU Loss, and the mAP reached 91.5%. Zhang et al [17]. proposed an insulator and defect dataset with synthetic Fog, and a algorithm based on YOLOv5 with channel attention mechanism and SE-CSP layer to effectively distinguish the features. The mAP0.5 and mAP0.5:0.95 reached 99.5% and 88.3%. These methods have great significance to insulator and defect detection, but there still remain some problems. (1) These methods have high detection accuracy on large objects like insulators, but have low detection rate on small defects. (2) Most of the methods have large number of parameters and a lot of floating-point operations (FLOPs), large model size, and low detection speed. (3) Some light-weight models can efficiently reduce size and parameters of the model, but they cannot achieve a balance between weight and accuracy.

Therefore, this paper proposed a light-weight insulator and defect detection network based on YOLOv5s, which can maintain high detection speed and accuracy. Our contributions are as follows:

- Ghost module [18] is introduced into YOLOv5 backbone and neck to reduce the parameters and model size.
- Small object detection network is proposed to increase detection rate of small insulator defects.
- CBAM [19] is applied to backbone of the network to select critical features of insulators and defects, and suppress the uncritical features to improve accuracy of the network.

2. The Architecture of Original Network

YOLOv5, proposed by Glenn Jocher was a state-of-the-art object detection algorithm. As one-stage algorithm, compare to the two-stage algorithm, is faster and more suitable for real-time insulator defect detection in UAVs. The YOLOv5 network structure contains backbone, neck, and head, as shown in Figure 1.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 11 May 2023 doi:10.20944/preprints202305.0796.v1

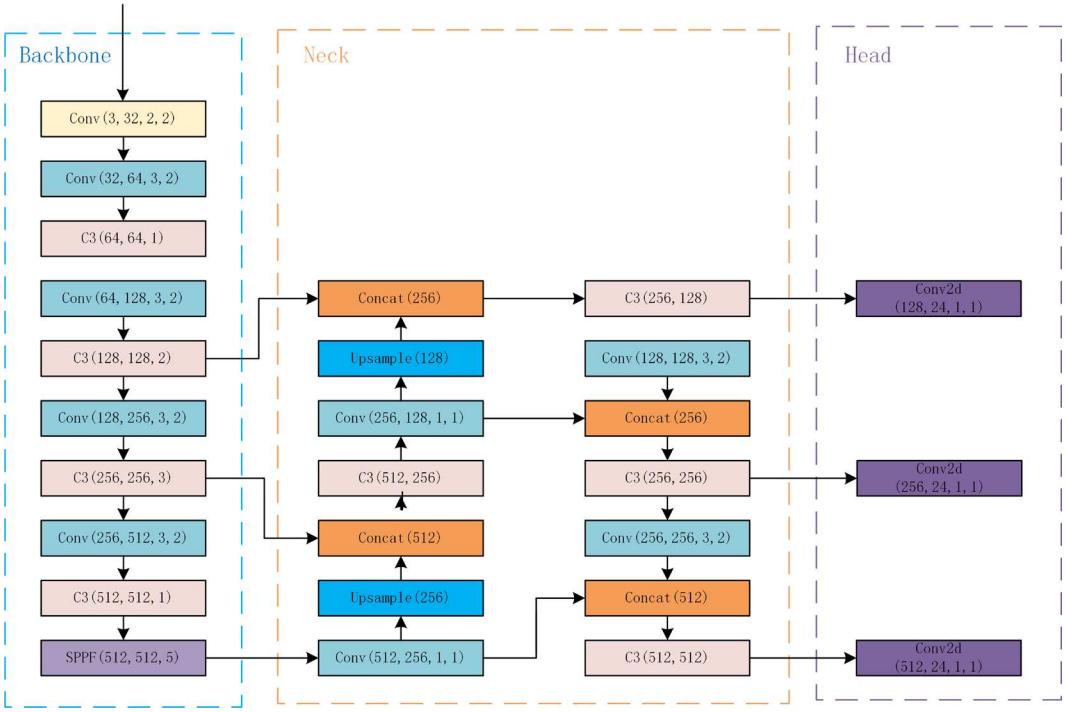


Figure 1. The network structure of YOLOv5.

2.1. Backbone

The improved CSP-Darknet53 is used in YOLOv5 as the backbone. Darknet53 refers to the structure of ResNet and adds residual blocks to the network architecture. It increases the depth of network while controlling the gradient propagation and avoiding gradient exploding or gradient vanishing. The C3 block was proposed to replace the original CSP block. It removed the Conv block after the residual output, and changed LeakyRelu to SiLU [20] to reduce the FLOPs.

2.2. Neck

The network structure of Neck follows the structure of FPN (Feature Pyramid Network) [21] with PAN (Path Aggregation Network) [22] to combine and transfer features in local (low-level) and global (high-level). The feature map carries richer semantic features and weaker localization features on higher level, and it carries richer localization features and weaker semantic features on lower level. The FPN helps to combine the rich semantic features from higher levels to lower levels. In contrast, PAN transmits the rich localization features from lower levels to higher levels. With the combination of PAN and FPN, the parameters are aggregated from the backbone layer and improve the capability of feature extraction.

2.3. Head

The head is three detection layers that make predictions based on the feature maps generated by the neck. It has three detection layers which consist three sizes of feature maps for different sizes of the object. It outputs the bounding boxes, the classification of the object, and the confidence score.

3. Method

3.1. Improved YOLOv5 Method

The pretrained weights include YOLOv5n, YOLOv5s, YOLOv5m, YOlOv5l and YOLOv5x. In comparison, the mAP of YOlOv5s is lower than YOLOv5x, but the parameters and FLOPs are much fewer than YOLOv5x and it is more adaptive for deployment to embedded devices like UAVs.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 11 May 2023 doi:10.20944/preprints202305.0796.v1

Therefore, we chose YOLOv5s as the pretrained weights for our model. Besides, we selected the latest version, YOLOv5v6. 2 as the benchmark. The Focus Layer was replaced with an equivalent 6x6 convolutional layer, and the latter has better exportability. Besides, Spatial Pyramid Pooling-Fast (SPPF) layer was proposed to replace Spatial Pyramid Pooling (SPP) [23] layer. SPP layer contains three different max-pooling layers in parallel connection as shown in Figure 2, to avoid shape and incomplete cropping distortion of images caused by the R-CNN algorithm. And it solves the problem of multiple feature extraction to improve the speed of generating bounding boxes. The architecture of the SPPF layer is connected by three same max-pooling layers in sequence, as shown in Figure 3. And SPPF can perform the same functions as SPP in half the time.

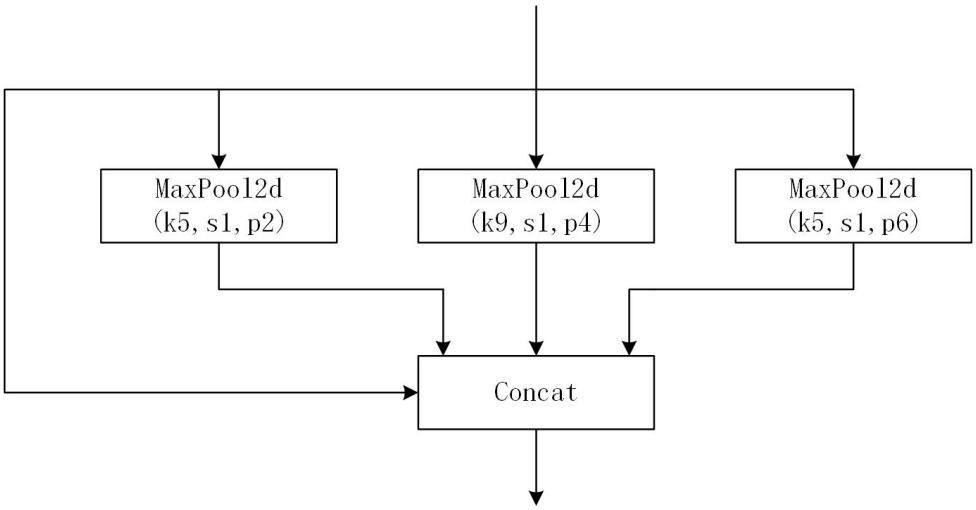


Figure 2. The architecture of the SPP layer.

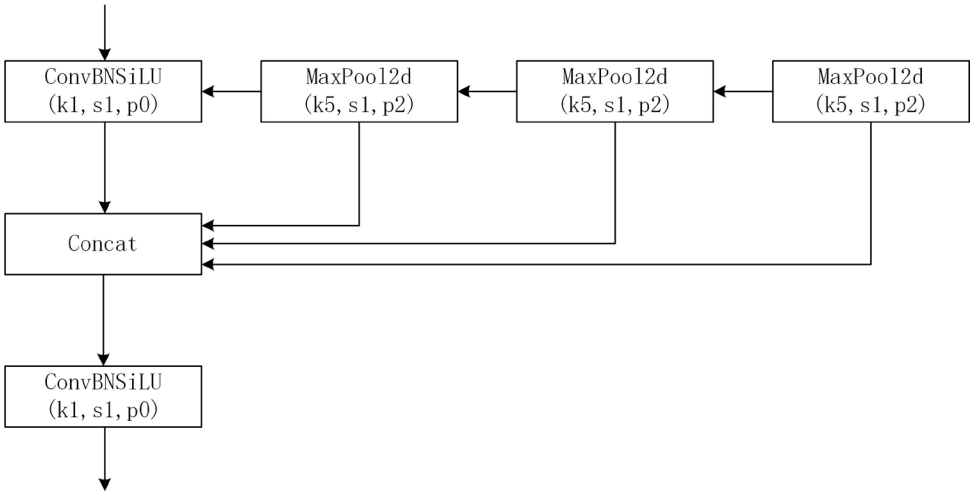
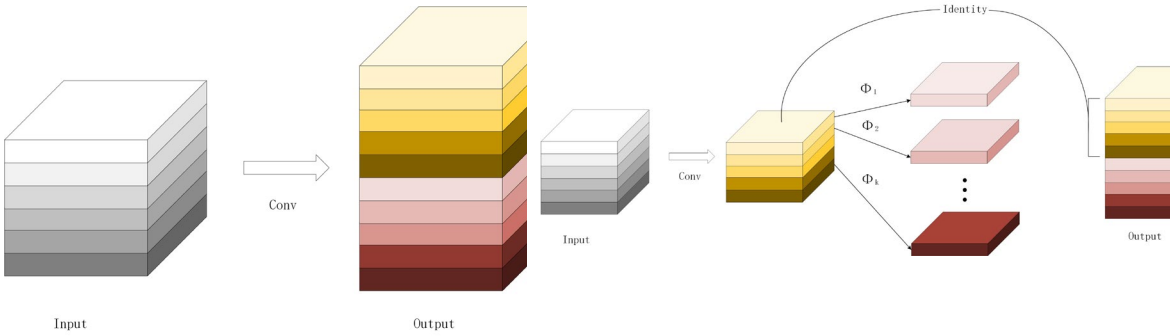


Figure 3. The architecture of the SPPF layer.

3.2. Lightweight Network with Ghost Module

Ghost module is a model compression method. It can extract more feature maps with smaller number of parameters to raise the computing speed and reduce latency in UAVs. The working process of the Ghost module is shown in Figure 4.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 11 May 2023 doi:10.20944/preprints202305.0796.v1



(a) (b)

Figure 4. The working process of (a) original convolutional layer and (b) Ghost module.

From the original convolutional layer, the input feature map is $h \times w \times c$ and the input kernel size is $k \times k$, where h is the input feature map height and w is the input feature map width, and c is the input channel numbers. It outputs the $h' \times w' \times n$ feature map, where h' is the output feature map height and w' is the output feature map width, and n is the output channel numbers. During original convolution process, FLOPs numbers required to be computed as $n \cdot h' \cdot w' \cdot c \cdot k \cdot k$, which the number of filters n is often over 10^5 and the number of channels c is 256 or 512. Ghost module extracts ghost feature maps by applying some cheap transformation operations and generates some intrinsic feature maps by using fewer number of filters. In Ghost module computation, the FLOPs can be computed as $(s-1) \cdot n/s \cdot h' \cdot w' \cdot d \cdot d + n/s \cdot h' \cdot w' \cdot c \cdot k \cdot k$, the $d \times d$ is equal to each linear operation averaged kernel size and s is the cheap transformation operation numbers and s is much smaller than c.

$$r_s = \frac{h' \cdot w' \cdot c \cdot k \cdot k \cdot n}{(s-1) \cdot \frac{n}{s} \cdot h' \cdot w' \cdot d \cdot d + \frac{n}{s} \cdot h' \cdot w' \cdot c \cdot k \cdot k} = \frac{c \cdot k \cdot k}{\frac{1}{s} \cdot c \cdot k \cdot k + \frac{s-1}{s} \cdot d \cdot d} \approx \frac{s \cdot c}{s + c - 1} \approx s \quad (1)$$

From formula (1), FLOPs required for the original convolutional layer are s times that of the Ghost module, and the parameters for computation can also be approximately equal to s. This shows the advantages of Ghost module in lightweight network. Therefore, GhostBottleneck and C3Ghost are proposed to further optimize network parameters [24]. Figure 5 shows the architecture of GhostBottleneck and C3Ghost. GhostBottleneck is designed in place of the bottleneck in original convolution, and C3Ghost is designed to replace the C3 layer. We replaced all original convolutional layers with Ghost convolutional layers and replaced all C3 layers with C3Ghost layers to reduce the FlOPs and computation parameters to make it better to meet the real-time detection requirement and easier to deploy to UAVs.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 11 May 2023 doi:10.20944/preprints202305.0796.v1

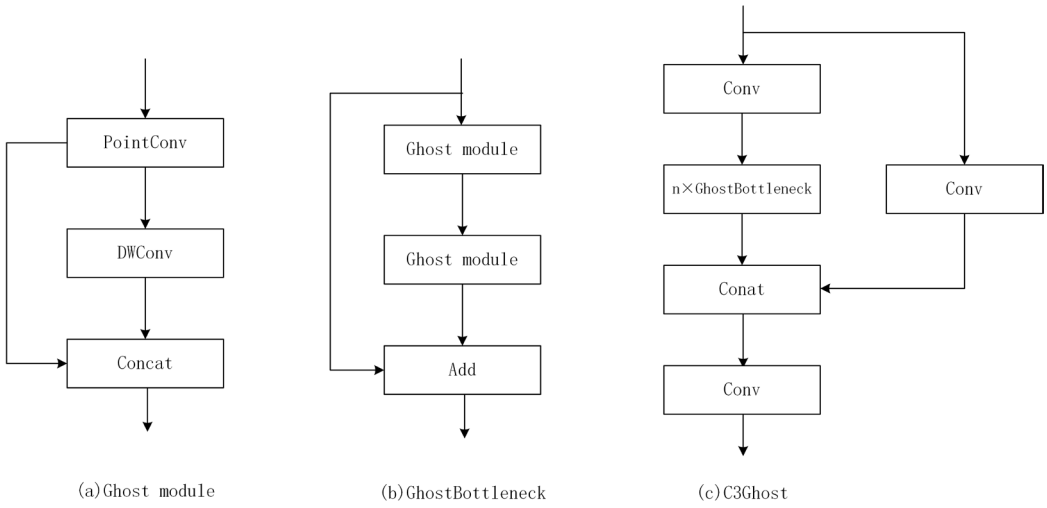


Figure 5. The architecture of Ghost module, GhostBottleneck and C3Ghost.

3.3. CBAM Attention Mechanism

To make the algorithm be able to focus on the most critical information through the inputs, decrease the attention to other useless information, we introduce CBAM to our network.

CBAM is a widely used and lightweight attention mechanism, which can be divided into CAM (channel attention module) and SAM (spatial attention module). When input the feature map $F \in R^{C\times H\times W}$, The CBAM infers one-dimensional channel attention feature map $M_C \in R^{C\times 1\times 1}$, and two-dimensional spatial attention feature map $M_S \in R^{1\times H\times W}$, as shown in Figure 6. The process of CBAM can be illustrated as:

$$F' = F \otimes M_c(F) \tag{2}$$

$$F'' = F' \otimes M_s(F') \tag{3}$$

Where C is the feature map channel, H is the feature map height and W is the feature map width, and $\otimes$ is element multiplication. F'' is final output.

The channel attention focuses on key channels by multiplying weights from different channels. The channel attention map $M_C \in R^{C\times 1\times 1}$ is compute as:

This section may be divided by subheadings. It should provide a concise and precise description of the experimental results, their interpretation, as well as the experimental conclusions that can be drawn.

$$M_c(F) = \sigma(W_1(W_0(F_{avg}^c)) + W_1(W_0(F_{max}^c))) \tag{4}$$

Where σ is the sigmoid activation function, W_1 and W_0 are the weights shared their inputs, and F_{avg}^c and F_{max}^c are the features of average-pool and features of max-pool.

The spatial attention learns spatial features by focusing on location information and assign weights for spatial features. The spatial features can be calculated as:

$$M_s(F) = \sigma(f^{7\times 7}([F_{avg}^s; F_{max}^s])) \tag{5}$$

Where $f^{7\times 7}$ is the convolution with 7×7 filter size.

As shown in Figure. 6, we introduced CBAM to the backbone of our improved network structure to focus on insulator and defect information, and suppress other useless information and increase the detection rate and efficiency of insulator detection.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 11 May 2023 doi:10.20944/preprints202305.0796.v1

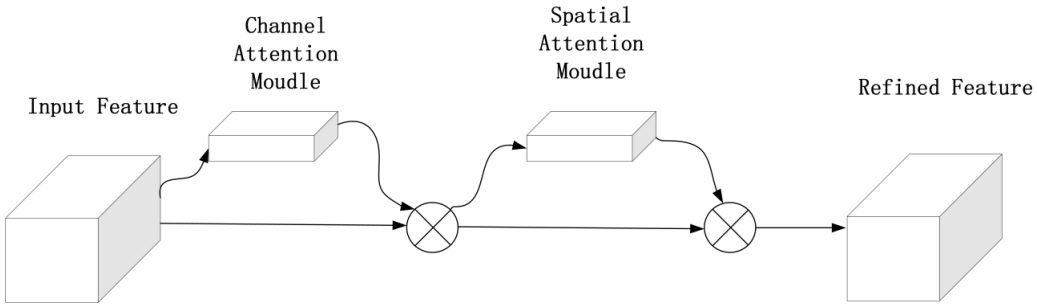


Figure 6. The CBAM block.

3.4. Small Object Detection Network

In the pictures taken by UAVs, the proportion of insulators and defects in the whole pictures is very small. We chose to change the backbone and neck structure to improve the small object detection accuracy. The head network is aimed to predict classes and bounding boxes from the feature maps extracted from the neck. It has little impact on small object detection. So, we only increased the numbers of the detection head.

In backbone, we redirected the connection of bottom layers, which include feature maps in higher resolution to send these feature maps directly into the neck. And we expanded the neck to adapt to extra feature maps. The structure of our network is shown in Figure. 7. In addition, we changed the sizes and numbers of the anchors applied in head and changed the parameters to be adjust to this new structure.

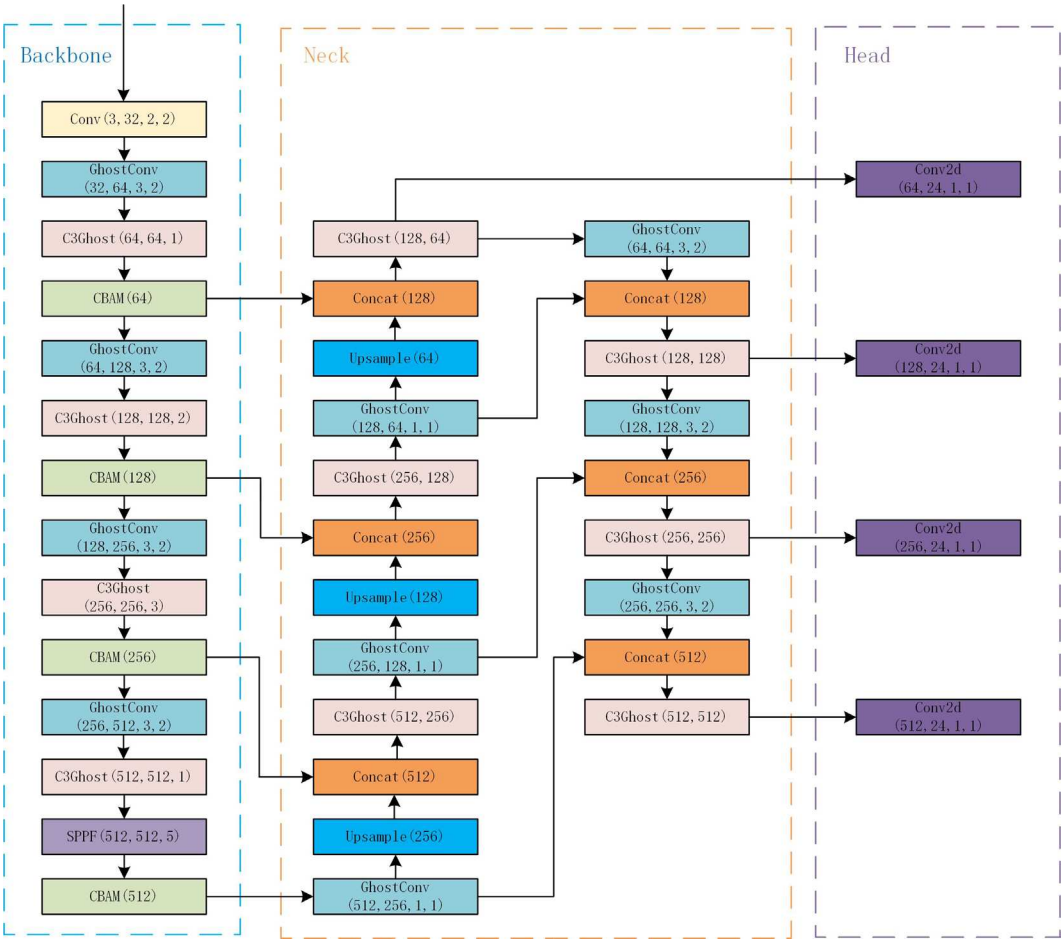


Figure 7. The overall network structure.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 11 May 2023 doi:10.20944/preprints202305.0796.v1

4. Experiments

4.1. Experiment Introduction

4.1.1. Dataset Description

Our dataset used two public insulator datasets: Chinese Power Line Insulator Dataset (CPLD) [28], provided by the State Grid Corporation of China. And Synthetic foggy insulator dataset (SFID), made by Zheng-De Zhang. And one insulator dataset from the Internet.

CPLD dataset contains 600m insulator images and 248 insulator images with defects partly collected by UAVs and partly generated by data augmentation method. And the SFID was based on CPLD and Unifying Public Datasets for Insulator (UPID). It constructed the dataset by applying synthetic fogging algorithm, which contains 13718 insulator and defect images. The dataset from the internet contains 678 insulator and defect images. We combined and filtered these three types of insulator datasets. Our dataset includes 7582 images in total and we divided the dataset into 7:2:1. 5307 images for training, 1517 images for verification, and 758 images for testing. The pixels of images are set into 1152x864 and stored into JPG format and the labeling files saved as VOC format.

4.1.2. Experimental Configuration

All experiments are in Linux operating system, the CPU is AMD Ryzen 5, and the GPU is NVIDIA RTX2080Ti with 11G memory. The software environment contains CUDA 11.2, CUDNN 8.40, Python 3.10 and PyTorch framework.

4.1.3. Evaluation Indicators

To test the performance of our proposed small insulator and defect detection model, we chose precision, recall, mAP0.5, mAP0.5:0.95, FLOPS, parameters and model size as evaluation indicators.

Precision is the proportion that the model predicted positive sample numbers correctly to the model predicted as positive sample numbers. It can be defined as:

$$Precision = \frac{T_P}{T_P + F_P} \tag{6}$$

Recall is the proportion that the model correctly predicted. It can be defined as:

$$Recall = \frac{T_P}{T_P + F_N} \tag{7}$$

Where the T_p is the positive sample numbers the model predicted as positive. F_P are the positive sample numbers the model predicted as positive. F_N are the positive sample numbers the model ignored.

The formulas of mAP0.5 and mAP0.5:0.95 can be defined as:

$$AP = \int_0^1 P(R)dR \tag{8}$$

$$mAP = \frac{\sum_{i=1}^{N} AP_1}{N} \tag{9}$$

Where mAP0.5 is the mean average precision when IoU is set into 0.5, and mAP0.5:0.95 is the mean average precision when IoU is set from 0.5 to 0.95 with 0.05 steps.

4.2. Experiment on Insulators and Defects

4.2.1. Experiment results

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 11 May 2023 doi:10.20944/preprints202305.0796.v1

We chose several experiments to verify the performance of the model in different network structures. There are four experiments trained in this article, YOLOv5s, YOLOv5s + Ghost module, YOLOv5s + Ghost module + small object detection network, YOLOv5s + Ghost module + small object detection network + CBAM. Table 1 shows the training results in different network structures. Table 2 shows the deploy performance for UAVs.

Table 1. Detection results of insulators and defect in different network structures

Model	Classes	Precision	Recall	mAP0.5	mAP0.5:0.95
YOLOv5s	Average	99.4%	99.3%	99.5%	90.4%
	Insulator	99.2%	98.8%	99.4%	93.1%
	Defect	99.6%	99.7%	99.5%	87.6%
YOLOv5s + Ghost module	Average	98.4%	98.5%	99.2%	89.1%
	Insulator	97.2%	97.4%	99.1%	92.0%
	Defect	99.6%	99.6%	99.3%	86.1%
YOLOv5s + Ghost module + small object detection network	Average	98.6%	98.9%	99.3%	91.2%
	Insulator	97.2%	98.1%	99.2%	91.6%
	Defect	99.9%	99.7%	99.5%	90.8%
YOLOv5s + Ghost module + small object detection network + CBAM	Average	98.7%	98.9%	99.4%	91.7%
	Insulator	97.9%	98.0%	99.3%	92.5%
	Defect	99.6%	99.7%	99.5%	90.8%

Table 2. Deploy performance in different network structures

Model	Parameters	FLOPs(G)	Weight(M)	Speed-GPU(ms/image)
YOLOv5s	7025025	16.0	13.72	9.5
YOLOv5s + Ghost module	3687239	8.2	7.44	9.3
YOLOv5s + Ghost module + small object detection network	3763460	9.8	8.69	10.5
YOLOv5s + Ghost module + small object detection network + CBAM	3807372	9.9	8.79	10.9

4.2.2. Influence of Ghost module

To reduce the model size and increase the performance of our model for UAVs, we introduced Ghost module to the network structure. In Table 2, the parameters of the model with Ghost module are reduced by 47.51% and the FLOPs are reduced by 48.75% compared with YOLOv5s. The model size is reduced by 45.77% and the processing speed is slightly improved in GPU. However, the average accuracy decreases by 0.3% in the evaluation of mAP0.5 and 1.3% in the evaluation of mAP0.5:0.95 in Table 1. For insulator, the mAP0.5 decreases by 0.3% and mAP0.5:0.95 decreases by 1.1%. For defect, the mAP0.5 decreases by 0.2% and mAP0.5:0.95 decreases by 1.5%. Ghost module makes it easier to deploy into UAVs and increase the speed of prediction, but these are all at the expense of accuracy.

4.2.3. Influence of small object detection network

To solve decreasing accuracy of Ghost module, we chose to change network structure to increase the accuracy. Table 1 shows that the average accuracy increases by 0.1% in the evaluation of mAP0.5 and 2.1% in the evaluation of mAP0.5:0.95 compared with YOLOv5s + Ghost module. For defect, the mAP0.5 increases by 0.2% and mAP0.5:0.95 increases by 4.7%. Although the deploy performance is

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 11 May 2023 doi:10.20944/preprints202305.0796.v1

slightly inferior to YOLOv5s + Ghost module, it is much superior to YOLOv5s and has significant improvement in small defect detection. accuracy decreases by 0.3% in the evaluation of mAP0.5 and 1.3% in the evaluation of mAP0.5:0.95 in Table 1. For insulator, the mAP0.5 decreases by 0.3% and mAP0.5:0.95 decreases by 1.1%. For defect, the mAP0.5 decreases by 0.2% and mAP0.5:0.95 decreases by 1.5%. Ghost module makes it easier to deploy into UAVs and increase the speed of prediction, but these are all at the expense of accuracy.

4.2.4. Influence of CBAM

To further improve the model accuracy, CBAM was proposed to our network structure. Table 1 shows the average accuracy increases by 0.1% in the evaluation of mAP0.5 and 0.5% in the evaluation of mAP0.5:0.95 compared with YOLOv5s + Ghost module + small object detection network. For insulators, the mAP0.5 increases by 0.1%, and mAP0.5:0.95 increases by 0.9%.

4.3. Comparison with Different Methods

We select some classic models and state-of-the-art models for comparison to verify the accuracy and performance of our model, which include Faster R-CNN, YOLOv3 and YOLOv4. All of the networks were trained on our proposed dataset. Table 3 shows the results of network performance.

As shown in Table 3, classic method faster R-CNN has poor performance on insulator and defect detection and UAV deployment. The speed of YOLOv3 and YOLOv4 are slightly faster than ours, but the mean average precision is much lower than our model. Our model maintains the high accuracy of YOLOv5s and greatly improve the capability of small defect detection. As shown in Figure. 8. Besides, we halve its parameters and weights to make it easier to deploy to embedded systems. The experiment results show that our model has better superiority and applicability though comparison.

Table 3. Comparison of different network performance

Model	mAP0.5	mAP0.5:0.95	Parameters	FLOPs(G)	Weight(M)	Speed-GPU(ms/image)
Faster R-CNN	97.2%	77.8%	19546215	7.8	74.25	8.8
YOLOv3	98.8%	79.7%	8654686	12.8	16.68	8.7
YOLOv4	99.2%	83.5%	8787543	16.5	11.34	9.1
YOLOv5s	99.5%	90.4%	7025025	16.0	13.72	9.5
Ours	99.4%	91.7%	3807372	9.9	8.79	10.9

Figure 8. Insulator and defect detection using YOLOv5s and our model under different conditions

5. Conclusions

This article proposes a light-weight insulator and defect detection network based on the improved YOLOv5, which meets the requirement of industrial deployment. The conclusions are as follows:

- Ghost module is introduced to the network structure of YOLOv5, which greatly decreases the parameters and FLOPs of the network, reduces model size by half, and maintains high speed of detection.
- Applying CBAM module can increase insulator detection accuracy with only a small increase in model weight and computation cost.
- The network changes for small object detection makes it easier to detect small defects and greatly increase the mean average precision of defect detection.

Through the comparison of experiment results, our model has better predictive performance and deploy performance compared with classic and state-of-the-art models. The average mAP0.5 can reach 99.4% and average mAP0.5:0.95 can reach 91.7%. And the speed of detection is 10.9ms/image, which can fully meet the real-time requirement. Moreover, the parameters and model weight decrease into 3807372 and 8.79M. Its light model weight and low computation cost make it easier to deploy to embedded devices like UAVs and reduce industrial cost.

Author Contributions: Conceptualization, Y.Z. and T.Z.; methodology, Y.Z. and Q.X; software, Y.Z. and J.H; validation, Y.Z., T.Z. and M.X.; formal analysis, Y.Z.; investigation, Y.Z.; resources, Y.Z. and J.L; data curation, Y.Z.; writing—original draft preparation, Y.Z. and T.Z.; writing—review and editing, Y.Z. and T.Z.; visualization, Y.Z.; supervision, Y.Z. and T.Z.; project administration.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Vaillancourt, G.H.; Carignan, S.; Jean, C. Experience with the detection of faulty composite insulators on high-voltage power lines by the electric field measurement method. *IEEE Transactions on Power Delivery* **1998**, *13*, 661-666, doi:10.1109/61.660958.
2. Yang, L.; Fan, J.; Liu, Y.; Li, E.; Peng, J.; Liang, Z. A Review on State-of-the-Art Power Line Inspection Techniques. *IEEE Transactions on Instrumentation and Measurement* **2020**, *69*, 9350-9365, doi:10.1109/TIM.2020.3031194.
3. Tao, X.; Zhang, D.; Wang, Z.; Liu, X.; Zhang, H.; Xu, D. Detection of Power Line Insulator Defects Using Aerial Images Analyzed With Convolutional Neural Networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **2020**, *50*, 1486-1498, doi:10.1109/TSMC.2018.2871750.
4. Zhai, Y.; Wang, D.; Zhang, M.; Wang, J.; Guo, F. Fault detection of insulator based on saliency and adaptive morphology. *Multimedia Tools and Applications* **2017**, *76*, 12051-12064, doi:10.1007/s11042-016-3981-2.
5. Arkin, E.; Yadikar, N.; Xu, X.; Aysa, A.; Ubul, K. A survey: object detection methods from CNN to transformer. *Multimedia Tools and Applications* **2022**, doi:10.1007/s11042-022-13801-3.
6. Cortes, C.; Vapnik, V. Support-vector networks. *Machine Learning* **1995**, *20*, 273-297, doi:10.1007/BF00994018.
7. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. In Proceedings of the Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 8-14 Dec. 2001, 2001; pp. I-I.
8. Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object Detection with Discriminatively Trained Part-Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2010**, *32*, 1627-1645, doi:10.1109/TPAMI.2009.167.
9. Zhang, S.; Wang, X. Human detection and object tracking based on Histograms of Oriented Gradients. In Proceedings of the 2013 Ninth International Conference on Natural Computation (ICNC), 23-25 July 2013, 2013; pp. 1349-1353.
10. Li, B.; Wu, D.; Cong, Y.; Xia, Y.; Tang, Y. A Method of Insulator Detection from Video Sequence. In Proceedings of the 2012 Fourth International Symposium on Information Science and Engineering, 14-16 Dec. 2012, 2012; pp. 386-389.
11. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, 23-28 June 2014, 2014; pp. 580-587.

12. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), 7-13 Dec. 2015, 2015; pp. 1440-1448.
13. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2017**, *39*, 1137-1149, doi:10.1109/TPAMI.2016.2577031.
14. Wang, Y.; Li, Z.; Yang, X.; Luo, N.; Zhao, Y.; Zhou, G. Insulator Defect Recognition Based on Faster R-CNN. In Proceedings of the 2020 International Conference on Computer, Information and Telecommunication Systems (CITS), 5-7 Oct. 2020, 2020; pp. 1-4.
15. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 27-30 June 2016, 2016; pp. 779-788.
16. Yang, Z.; Xu, Z.; Wang, Y. Bidirection-Fusion-YOLOv3: An Improved Method for Insulator Defect Detection Using UAV Image. *IEEE Transactions on Instrumentation and Measurement* **2022**, *71*, 1-8, doi:10.1109/TIM.2022.3201499.
17. Zhang, Z.D.; Zhang, B.; Lan, Z.C.; Liu, H.C.; Li, D.Y.; Pei, L.; Yu, W.X. FINet: An Insulator Dataset and Detection Benchmark Based on Synthetic Fog and Improved YOLOv5. *IEEE Transactions on Instrumentation and Measurement* **2022**, *71*, 1-8, doi:10.1109/TIM.2022.3194909.
18. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More Features From Cheap Operations. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 13-19 June 2020, 2020; pp. 1577-1586.
19. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the Computer Vision – ECCV 2018, Cham, 2018//, 2018; pp. 3-19.
20. Elfwing, S.; Uchibe, E.; Doya, K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks* **2018**, *107*, 3-11, doi:https://doi.org/10.1016/j.neunet.2017.12.012.
21. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 21-26 July 2017, 2017; pp. 936-944.
22. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 18-23 June 2018, 2018; pp. 8759-8768.
23. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2015**, *37*, 1904-1916, doi:10.1109/TPAMI.2015.2389824.
24. Dong, X.; Yan, S.; Duan, C. A lightweight vehicles detection network model based on YOLOv5. *Engineering Applications of Artificial Intelligence* **2022**, *113*, 104914, doi:https://doi.org/10.1016/j.engappai.2022.104914.