Proceedings of Machine Learning Research vol 278:1–8, 2025 2025 2nd International Conference on Machine Learning and Intelligent Computing

# YOLOv5n-ShuffleNetv2: A Lightweight Transmission Line Insulator Defect Detection Algorithm

**Wanbo Luo** BOBO504@126.COM
*Leshan Vocational and Technical College, Leshan 614000, China*

**Editors:** Nianyin Zeng, Ram Bilas Pachori and Dongshu Wang

## Abstract

Insulators represent a pivotal component of transmission lines, with their functionality directly impacting the reliable operation of the power grid. Current insulator defect detection algorithms face significant challenges, including low accuracy and long latency, which hinder their practical application in the timely and reliable maintenance of power systems. Therefore, this paper proposed a lightweight detector to minimize the model's parameters and calculations. First, the YOLOv5n algorithm was chosen as the foundation for the detection system's lightweight design. Second, the ShuffleNetv2 backbone replaced the YOLOv5n backbone to further lightweight the model. Experimental results showed that the proposed YOLOv5n-ShuffleNetv2 model achieved 84.5% mAP50 at 87.7 FPS using the Nvidia Jetson Orin Nano 4G. Although the accuracy decreased by 4.1%, the model achieved a 44.7% reduction in the number of parameters and a 22% increase in detection speed, demonstrating a significant improvement in efficiency and practicality.

**Keywords:** Insulator Defect Detection, YOLOv5n, ShuffleNetv2, Unmanned Aerial Vehicle (UAV).

## 1. Introduction

Insulators play a critical role in transmission lines, supporting and electrically isolating conductors to prevent current leakage and ensure safety. Their performance directly impacts the stable operation of the power grid, as any failure or degradation can lead to power disruptions, reduced efficiency, or safety hazards (Zhang et al., 2021).

The primary approach to the identification of insulator defects in the past was based on the extraction of defect features by means of conventional computer image processing technology (Shakiba et al., 2022). Subsequently, object detection algorithms were developed that extracted object features based on artificial design methods. These algorithms then used classifiers in machine learning for object detection. However, the above techniques are labor-intensive and inefficient, and their applicable detection scenarios are limited.

Insulator defect detection algorithms based on deep learning automatically extract object features and have been widely used (Panigrahy and Karmakar, 2024). However, these algorithms need improvement for insulator defect detection in UAVs, which face limited computing and storage resources. A lightweight detection model balances efficiency and performance, enabling faster detection with minimal accuracy drop.

Therefore, this paper proposed a lightweight insulator defect detection algorithm. The main contributions of this paper are as follows: (1) The lightweight YOLOv5n algorithm in the YOLO series was selected to minimize the number of parameters and calculations of the model. (2) The YOLOv5n backbone is replaced with the ShuffleNetv2 backbone to further enhance the model's lightweight design.

## 2. Related Works

In 2022, MA and ZHANG (2022) enhanced Faster R-CNN for insulator detection by integrating SKNet for feature focus, replacing Batch Normalization (BN) with the Filter Response Normalization (FRN) to prevent gradient saturation, and using the distance intersection over union (DIoU) for better localization, achieving 89.79% average precision.

In 2019, Wu et al. (2019) proposed an improved YOLOv3-based insulator defect detection algorithm, using K-means++ for optimal anchor box sizing and replacing the YOLOv3 backbone with Crop-MobileNet to reduce parameters. The model achieved 84.0% mAP at 46.3 FPS using an Nvidia RTX 1080 Ti GPU.

In 2022, Wang et al. (2022) proposed an improved YOLOv5-based insulator detection method, incorporating the K3-Ghost structure to reduce parameters and the SENet module to enhance channel attention. The model achieved 89.5% mAP50 at 35.7 FPS using an Nvidia RTX 3060 Ti GPU.

Model lightweighting can be achieved by using smaller convolution kernels, but this may result in a significant decrease in accuracy. To alleviate the decrease in accuracy, multi-scale features are added to enhance feature fusion, but this will increase parameters. Meanwhile, to ensure the integrity of the network structure, replacing the original backbone with a lightweight backbone may be a good solution.

Despite high accuracy, existing models face challenges with large parameters and computational complexity. YOLOv3 has over 20 million parameters, and even the lightweight YOLOv5 (over 1.9 million) model remains resource-intensive, limiting their performance on UAVs.

## 3. Methodology

This paper proposed a lightweight insulator defect detector based on YOLOv5n. To further reduce the model's complexity while maintaining performance, the ShuffleNetv2 backbone replaced the original YOLOv5n backbone. This modification balanced accuracy and speed, making it suitable for resource-constrained UAV-based detection environments.

### 3.1. YOLOv5n

The YOLOv5n model is the smallest in the YOLOv5 series. Table 1 displays the overall architecture of the YOLOv5n backbone.

Table 1: Overall architecture of the ShuffleNetv2 backbone.

| Input | Operator | Output |
|---|---|---|
| $640 \times 640 \times 3$ | Conv | $320 \times 320 \times 16$ |
| $320 \times 320 \times 16$ | Conv | $160 \times 160 \times 32$ |
| $160 \times 160 \times 32$ | C3 | $160 \times 160 \times 32$ |
| $160 \times 160 \times 32$ | Conv | $80 \times 80 \times 64$ |
| $80 \times 80 \times 64$ | C3 | $80 \times 80 \times 64$ |
| $80 \times 80 \times 64$ | Conv | $40 \times 40 \times 128$ |
| $40 \times 40 \times 128$ | C3 | $40 \times 40 \times 128$ |
| $40 \times 40 \times 128$ | Conv | $20 \times 20 \times 256$ |
| $20 \times 20 \times 256$ | C3 | $20 \times 20 \times 256$ |
| $20 \times 20 \times 256$ | SPPF | $20 \times 20 \times 256$ |

The YOLOv5n backbone generated a feature map with a shape of 320 × 320 × 16 by a convolutional layer. Through four sets of operations (Conv and C3), the feature map generated 160 × 160 × 32, 80 × 80 × 64, 40 × 40 × 128, and 20 × 20 × 256 feature maps. The last three feature maps predicted small, medium, and large objects.

### 3.2. ShuffleNetv2

ShuffleNet is an efficient CNN for mobile and embedded devices, using Depthwise Separable Convolutions (DSC), Group Convolutions, and Channel Shuffle (CS) for improved speed and accuracy. Table 2 displays the overall architecture of the ShuffleNetv2 backbone.

Table 2: Overall architecture of the YOLOv5n backbone.

| Input | Operator | Output |
|---|---|---|
| 640 × 640 × 3 | Conv (3 × 3) | 320 × 320 × 24 |
| 320 × 320 × 24 | Max_Pooling (3 × 3) | 160 × 160 × 24 |
| 160 × 160 × 24 | ShuffleNet_Unit × 4 | 80 × 80 × 48 |
| 80 × 80 × 48 | ShuffleNet_Unit × 8 | 40 × 40 × 96 |
| 40 × 40 × 96 | ShuffleNet_Unit × 4 | 20 × 20 × 192 |

Where the shape of the feature map denotes length × width × channel. The ShuffleNetv2 backbone generated five feature maps with different shapes by downsampling the feature map with a shape of 640 × 640 × 3. First, the input feature map with a shape of 640 × 640 × 3 was passed through a convolutional layer to output the feature map with a shape of 320 × 320 × 24. Second, the shape of the output feature map was reduced to 160 × 160 × 24 by a Max_Pooling layer. Then, the feature map was passed through multiple ShuffleNet_Unit layers to produce three feature maps with the shapes of 80 × 80 × 48, 40 × 40 × 96, and 20 × 20 × 192 to predict small, medium, and large objects.

### 3.3. Backbone Improvement

For the YOLOv5n backbone to be effectively replaced, the ShuffleNetv2 backbone must generate three feature maps with dimensions of 80 × 80, 40 × 40, and 20 × 20, ensuring compatibility with the original model architecture. Fig. 1 shows the replacement of the YOLOv5n backbone with the ShuffleNetv2 backbone.

Layers 0 to 2 of ShuffleNetv2 replaced layers 0 to 4 of YOLOv5s to produce an 80 × 80 × 48 feature map. Similarly, layer 4 of ShuffleNetv2 replaced layers 5 to 6 of YOLOv5n to output a 40 × 40 × 96 feature map. Finally, layer 5 of ShuffleNetv2 replaced layers 7 to 8 of YOLOv5n to generate a 20 × 20 × 192 feature map. Therefore, the ShuffleNetv2 backbone reduced the channel number of each feature map for lightweight design. Furthermore, the DSC of the ShuffleNetv2 further reduced parameters compared to the Conv of the YOLOv5n backbone.

## 4. Results

The insulator defect dataset used in this paper includes 2,300 images, consisting of Internet-sourced and high-definition UAV-captured, with 1,280 used for training and 420 for validation. The GPU used for training is the Nvidia RTX 2060. The training epochs are set to 300. The trained model was

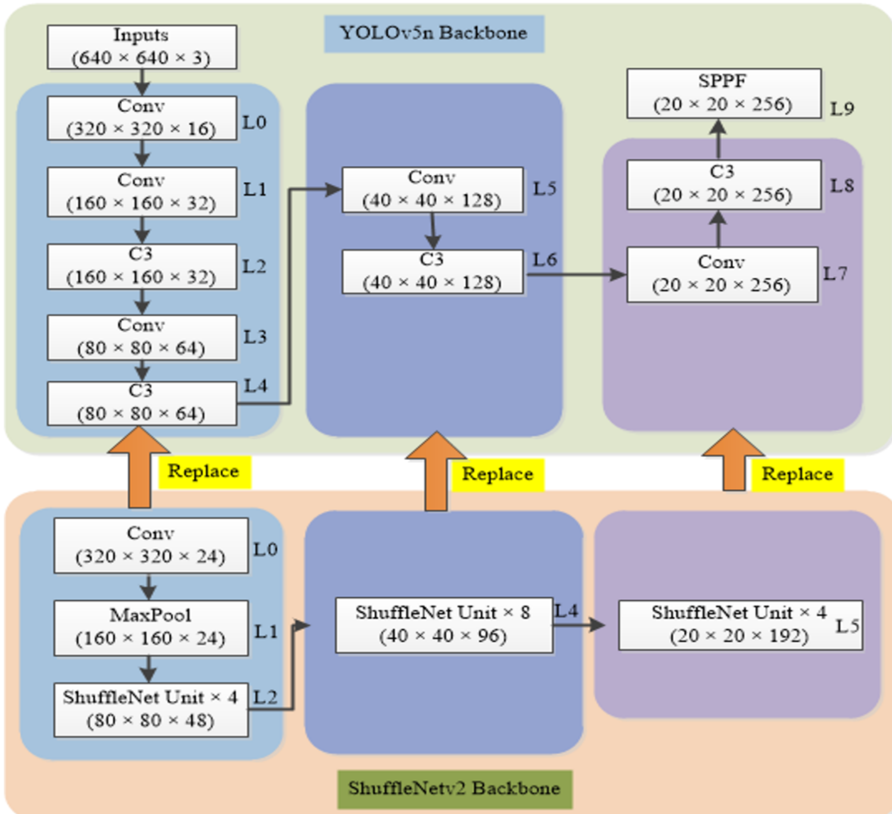


Figure 1: Backbone improvement.

deployed on a Jetson Orin Nano, simulating the main processor for the UAV's visual recognition tasks.

### 4.1. Training Curves

Fig. 2 shows the training curves of the YOLOv5n model. The box_loss, obj_loss, and cls_loss gradually dropped during the training period. During the validation period, the box_loss and obj_loss gradually decreased. However, the cls_loss fluctuated slightly. The recall, mAP_0.5, and mAP_0.5:0.95 metrics gradually increased during 200 training epochs. Conversely, the precision metric exhibited a decline in the initial 20 epochs, followed by a gradual increase in the subsequent 180 epochs.

Fig. 3 shows the training curves of the YOLOv5n-ShuffleNetv2 model. The training curve of the YOLOv5n-ShuffleNetv2 model was similar to that of the YOLOv5n model.

### 4.2. Validation Results

The validation results of the YOLOv5n model are shown in Table 3.

The YOLOv5n model, trained with the original YOLOv5n algorithm, achieved an overall precision of 89.8%, recall of 86.0%, and mAP_0.5 of 88.6%.

The validation results of the YOLOv5n-ShuffleNetv2 model are shown in Table 4.

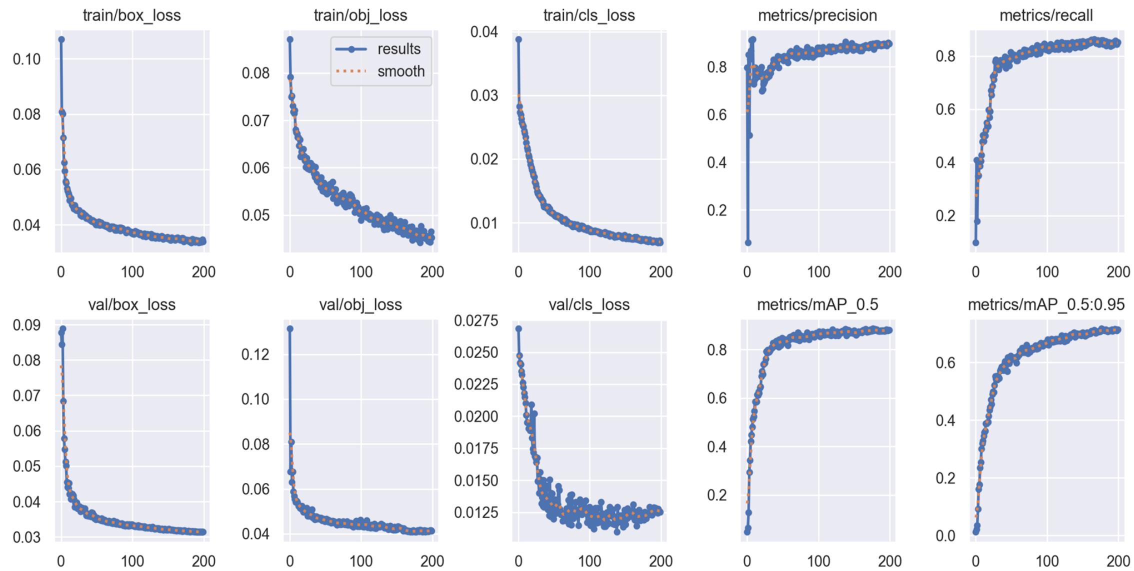


Figure 2: Training Curves of the YOLOv5n model.

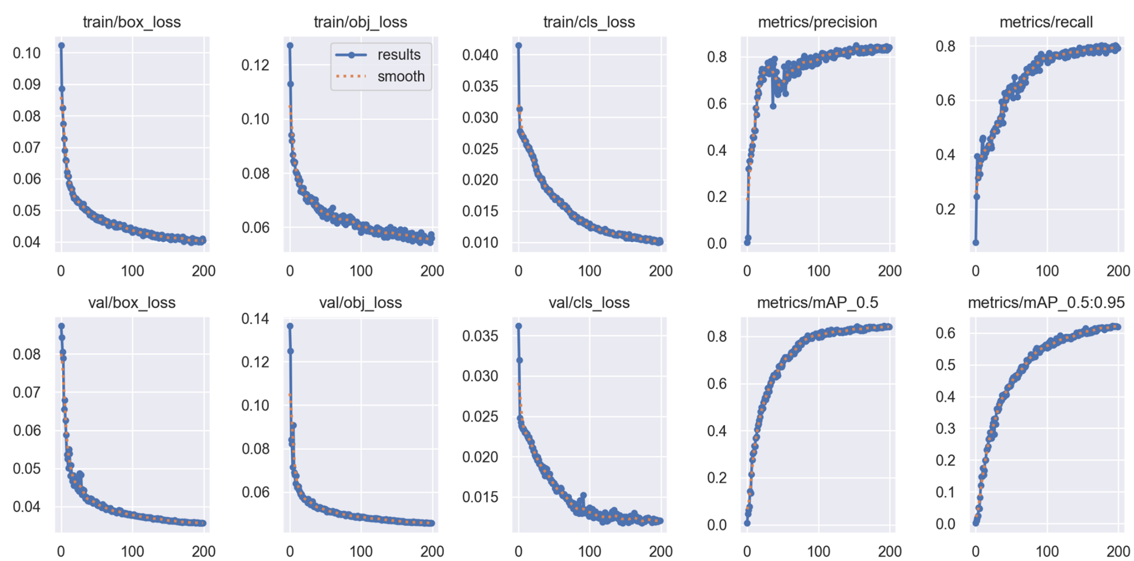


Figure 3: Training Curves of the YOLOv5n-ShffuleNetv2 model.

The YOLOv5n-ShuffleNetv2 model achieved a mAP_0.5 of 84.5% for all classes, with strong performance in the "string" class with 97.4% mAP_0.5 and "normal" class with 95.3% mAP_0.5. However, the "broken" class with 69.2% mAP_0.5 and "flashover" class with 76.0% mAP_0.5 showed lower accuracy, primarily due to reduced recall.

Table 3: Validation results of the YOLOv5n model.

| Class | Precision (%) | Recall (%) | mAP_0.5 (%) |
|---|---|---|---|
| all | 89.8 | 86.0 | 88.6 |
| string | 99.7 | 95.1 | 99.1 |
| normal | 94.8 | 93.4 | 95.8 |
| broken | 96.7 | 64.0 | 77.0 |
| flashover | 68.1 | 91.3 | 82.7 |

Table 4: Validation results of the YOLOv5n-ShuffleNetv2 model.

| Class | Precision (%) | Recall (%) | mAP_0.5 (%) |
|---|---|---|---|
| all | 83.0 | 79.8 | 84.5 |
| string | 94.2 | 89.5 | 97.4 |
| normal | 89.2 | 93.8 | 95.3 |
| broken | 84.2 | 54.9 | 69.2 |
| flashover | 64.2 | 80.8 | 76.0 |

### 4.3. Performance Comparison

Table 5 compares the performance of the YOLOv5n and YOLOv5n-ShuffleNetv2 models.

Table 5: Performance comparison.

| Model | Precision (%) | Recall (%) | mAP_0.5 (%) | Parameters |
|---|---|---|---|---|
| YOLOv5n | 89.8 | 86.0 | 88.6 | 1,764,577 |
| YOLOv5n-ShuffleNetv2 | 83.0 | 79.8 | 84.5 (-4.1%) | 975,953 (-44.7%) |

The YOLOv5n-ShuffleNetv2 model with 84.5% mAP50 underperformed the YOLOv5n model with 88.6% mAP50 but had a significantly lower parameter count, reducing the number of parameters by about 44.7% from 1,764,577 to 975,953.

Table 6 compares the detection speed of the YOLOv5n and YOLOv5n-ShuffleNetv2 models on the Jetson Orin Nano.

Table 6: Speed comparison of the two models.

| Model | Preprocess (ms) | Inference (ms) | NMS (ms) | FPS |
|---|---|---|---|---|
| YOLOv5n | 1.3 | 8.2 | 4.4 | 71.9 |
| YOLOv5n-ShuffleNetv2 | 1.3 | 5.8 | 4.3 | 87.7 |

Detection time includes preprocessing, inference, and non-maximum suppression (NMS) time. FPS is the inverse of the detection time. The YOLOv5n-ShuffleNetv2 model outperformed the YOLOv5n model in speed, achieving 87.7 FPS compared to 71.9 FPS, reflecting a 22% increase in detection speed.

### 4.4. Detection Results

Fig. 4 shows the detection results of the YOLOv5n-ShuffleNetv2 model.

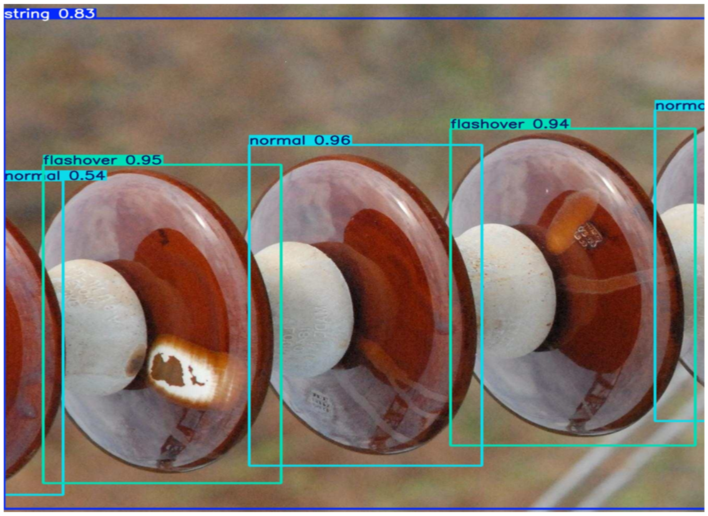


Figure 4: Detection results.

The YOLOv5n-ShuffleNetv2 model correctly detected the insulator string, normal and flashover insulators. The normal insulators shown in the figure achieved confidence values of 0.54 and 0.96. The flashover insulators shown in the figure achieved confidence values of 0.95 and 0.94. Its precise detection of these components enhanced the efficiency of power grid inspections, reduced the need for manual assessment, and supported proactive maintenance strategies in the electrical industry.

## 5. Conclusions

The YOLOv5n-ShuffleNetv2 model proposed an effective solution for detecting transmission line insulator defects, addressing the challenges of limited computational resources on UAV platforms. By replacing the YOLOv5n backbone with the ShuffleNetv2 backbone, the model reduced parameters by about 44.7% significantly while maintaining high performance with a slight decrease in accuracy from 88.6% to 84.5%. The improved model also demonstrated a notable 22% improvement in detection speed, achieving 87.7 FPS compared to YOLOv5n's 71.9 FPS, thus enhancing its practical applicability in real-time UAV-based inspections. This lightweight model is well-suited for integration into power grid maintenance systems, providing an efficient, scalable solution for insulator defect detection with minimal computational overhead.

## Acknowledgments

This research was funded by the Science and Technology Bureau of Leshan, Leshan, China (Grant number 24ZDYF0007).

## References

Yaoming MA and Yu ZHANG. Insulator detection algorithm based on improved faster-rcnn. *Journal of Computer Applications*, 42(02):631–637, 2022. doi: https://doi.org/10.11772/j.issn.1001-9081.2021020342.

Satyajit Panigrahy and Subrata Karmakar. Real-time condition monitoring of transmission line insulators using the yolo object detection model with a uav. *IEEE Transactions on Instrumentation and Measurement*, 73:1–9, 2024. doi: 10.1109/TIM.2024.3381693.

Fatemeh Mohammadi Shakiba, S. Mohsen Azizi, and Mengchu Zhou. A transfer learning-based method to detect insulator faults of high-voltage transmission lines via aerial images: Distinguishing intact and broken insulator images. *IEEE Systems, Man, and Cybernetics Magazine*, 8 (4):15–25, 2022. doi: 10.1109/MSMC.2022.3198027.

S. Wang, L. Zhao, M. Shao, and R. Ge. Insulator identification method of transmission line based on improved yolov5. *Electronic Measurement Technology*, 45(21):181–188, 2022. ISSN 1002-7300. doi: 10.19651/j.cnki.emt.2210118.

T. Wu, W. Wang, L. Yu, B. Xie, W. Yin, and H. Wang. Insulator defect detection method for lightweight yolov3. *Computer Engineering*, 45(08):275–280, 2019. doi: https://doi.org/10.19678/j.issn.1000-3428.0053695.

Zhaoyun Zhang, Guanfeng He, Shihong Huang, and Zhi Zhang. Application and development trend of target detection in power inspection. *Journal of Hubei Minzu University (Natural Science Edition)*, 39(3):305–314, 2021. doi: 10.13501/j.cnki.42-1908/n.2021.09.012.