An Enhanced Single-Stage Neural Network for Object Detection in Transmission Line Inspection

Changyu Cai¹, Jianglong Nie², Jie Tong¹,*, Zhao Chen², Xiangnan Xu¹ and Zhouqiang He²

¹ Artificial Intelligence Application Research Center, China Electric Power Research Institute, Beijing 100192, China; xuxiangnan@epri.sgcc.com.cn (X.X.)
² State Grid Gansu Electric Power Company, Lanzhou 730070, China; chenzhao@gs.sgcc.com.cn (Z.C.);
hezhq@gs.sgcc.com.cn (Z.H.)
* Correspondence: tongjie1@epri.sgcc.com.cn

Abstract: To address the issue of human object detection in transmission line inspection, an enhanced single-stage neural network is proposed, which is based on the improvement of the YOLOv7-tiny model. Firstly, a lighter GSConv module is utilized to optimize the original ELAN module, reducing the parameters in the network. In order to make the network less sensitive to the targets with an unconventional pose, a module based on CSPNeXt and GSConv is designed and integrated with the ELAN module to extract deep features from the targets. Moreover, a WIoU (Wise Intersection over Union) loss function is utilized to enhance the ability of the YOLOv7-tiny model to detect objects with an unconventional pose and the interference of the background. Finally, the experimental results on human targets in transmission line inspection demonstrate that the proposed network improves detection confidence and reduces missed detection. Compared to the YOLOv7-tiny model, the proposed method promotes the performance of accuracy while reducing the amount of parameters.

Keywords: object detection; transmission line inspection; YOLOv7-tiny; WIoU; GSConv

1. Introduction

With the rapid development of computer vision, visual monitoring has penetrated into every aspect of our lives. Human target detection [1,2], as an important branch of target detection, has attracted more and more attention in intelligent video surveillance [3]. Accordingly, neural network technologies as one of the most active fields of artificial intelligence provide a theoretical foundation for detecting targets, thus promoting monitoring efficiency, and reducing the burden of manual work [4].

Recently, deep learning neural networks have played a significant role in object detection [5,6], not only improving detection accuracy but also promoting detection speed. Generally, there exists three mainstream approaches, including one-stage [7–9] and two-stage methods [10–13], and the Transformer-based method [14,15]. The single-stage methods, exemplified by YOLO (You Only Look Once) algorithms [7], directly predict the bounding boxes and class probabilities in a single stage. This kind of method is renowned for its real-time performance and efficiency [16,17]. The two-stage methods appeared relatively early, dominated by CNN algorithms, such as Faster R-CNN and R-CNN series [13,14]. Usually, this method involves the generation of candidate boxes extracted from the image, and then the detection procedure is facilitated by making secondary adjustments based on these candidate regions. These kinds of methods can achieve high accuracy by explicitly generating region proposals and then classifying and refining them. The Transformer-based method is a state-of-the-art method which views object detection as a set prediction problem. However, it demands an extra-long training time to converge.

For years, much work has been devoted to object detection by using the aforementioned models. For instance, Yi et al. [18] proposed an enhanced algorithm called LAR-YOLOv8, which improves small target detection accuracy by designing an attention-guided...
bidirectional feature pyramid network, while introducing the RIoU loss function. Wang et al. [19] proposed an innovative anchor-free driving scene detection network based on YOLOv8, improving both speed and accuracy to meet the demands of high-speed road driving scenarios. Shi et al. [9] proposed the Maritime Tiny Person detector (MTP-YOLO) based on YOLOv9, which incorporates the C2fELAN design and MCE-CBAM module, along with a new W-EIoU loss function. For two-stage methods, He et al. [10] introduced an extension to the Faster R-CNN framework by adding a branch for predicting segmentation masks on each region of interest. Cai et al. [11] proposed enhancing the object detection framework through a multi-stage architecture where detection is progressively refined, improving the precision of bounding boxes at stages with incrementally higher intersection over union thresholds. Hu et al. [20] built on the Faster R-CNN by incorporating relation modules that model pairwise relations between objects in a scene, aiding in the recognition of contextually related objects. For Transformer-based methods, Meng et al. [12] introduced a method called conditional DETR (Detection Transformer) by incorporating a conditional spatial query. Zhu et al. [21] proposed an enhancement to the original DETR that addresses its slow convergence and limited feature resolution by using attention modules that focus on a small set of key sampling points. This adaptation significantly reduces training time and improves performance, particularly on small objects, as demonstrated on the COCO benchmark.

In contrast to the two other kinds of methods, YOLO series methods eliminate the need for separate region proposal generation, making them faster but possibly sacrificing some accuracy. In particular, the YOLO-tiny object detection model is designed for faster inference speed and lower resource requirements, making it suitable for running on mobile or device endpoints. However, it may cause a loss of precision, so that the unconventional pose and the interference of the background make the YOLO-tiny model have low performance on human detection in transmission line inspection.

To detect the human target in transmission line inspection, this paper proposes an enhanced single-stage neural network, which is based on the YOLOv7-tiny model [22]. The contributions of this work are as follows: (1) the introduction of a lighter GSConv module [23] to optimize the original ELAN module; (2) the combination of a CSPNeXt-GS module to make the network less sensitive to the target with an unconventional pose; and (3) the integration of the CSPNeXt-GS module to extract deep features from the targets. In addition, a WIoU loss function [24] is utilized to further enhance the ability of the YOLOv7-tiny model for object detection. By incorporating these improvements, the proposed method significantly promotes the model’s performance in terms of accuracy and faster convergence speed, making it a valuable contribution to the field of human target detection in transmission line inspection.

The organization of the paper is as follows: in Section 2, the original YOLO-tiny model is presented, and its modifications are described in Section 3. In Section 4, the experimental results of the proposed YOLO-tiny on human target detection in transmission line inspection are presented, and the performance of the proposed method is then compared with that of some existing methods to demonstrate its effectiveness in object detection. Finally, conclusions are drawn in Section 5.

2. YOLOv7-Tiny Neural Network

YOLOv7-tiny is a variant of the YOLOv7 object detection algorithm [25], which is designed for real-time and efficient object detection tasks. Generally speaking, it is optimized for edge devices with limited computational resources, thereby reducing the model size and computational complexity.

Figure 1 illustrates the overall structure of the YOLOv7-tiny model. It mainly consists of several key components, including a backbone network and detection layers. The backbone network is responsible for extracting features from the input image, which typically consists of several convolutional layers and pooling layers. The detection layers in YOLOv7-tiny model are utilized to generate predictions, taking the responsibility for
predicting bounding boxes, class probabilities, and objectless scores for different grid cells in the input image.

**Figure 1.** Original network structure of YOLOv7-tiny.

### 3. An Enhanced Single-Stage Neural Network Based on the YOLOv7-Tiny Model

The network architecture of an enhanced YOLOv7-tiny model is illustrated in Figure 2. Several improvements have been made compared to the original YOLOv7-tiny model, as seen in the boxes with color in Figure 2. Firstly, the improved ELAN-CSPGS module is used in the backbone network to enhance the extraction of deep features from the image. Secondly, the improved ELAN-GS module is employed in the network head to preserve the hidden connections between feature map channels as much as possible. The standard convolution in SPPCSPC-tiny is then replaced with GSCConv [23] to further reduce the number of parameters inherent in the network, and named SPPCSPC-GS for simplification. Finally, the network utilizes WIoU loss [24] to train the model, regardless of the original loss function.

**Figure 2.** Structure of an enhanced neural network.
In the following subsections, a detailed description of each module will be provided.

3.1. ELAN-GS Module

Figure 3 illustrates the original ELAN-tiny structure and its improvement structure. In contrast to the original ELAN-tiny model, a GSConv module is introduced into the ELAN-tiny model, regardless of convolutional blocks (CBL) which consists of convolutional layer, batch normalization (BN) and LeakyReLU activation function. The principle of GSConv is shown in Figure 4, where “Conv” represents standard convolution operations followed by batch normalization and a non-linear activation function, and “DWConv” represents depth-wise separable convolution which is a commonly used method to reduce the number of model parameters.

Let the input feature map information be $I$ and the output feature map be $O$. After, the ELAN-GS module operation can be expressed as

$$O = g^3_1(Concat(g^1_1(I), g^2_1(I), T^3_1(g^2_1(I)), T^3_3(l_0)))$$

(1)

where $T^i_n$ represents the standard convolution, pooling, and activation operation for the $i$-th convolution kernel of size $n$; $g^i_n$ is the GSConv operation for the $i$-th convolution kernel of size $n$; $Concat$ denotes splicing feature vectors along the channel dimension; $l_0 = T^3_3(g^2_1(I))$.

Similar to the original ELAN-tiny model, the input feature map undergoes two GSConv operations to extract preliminary features. Then, one of the GSConv outputs is subjected to two consecutive standard convolutions. Finally, the outputs of the four convolution blocks are concatenated, followed by a convolution fusion to obtain the final output. This process ensures that the final output $O$ contains more and deeper features.
3.2. ELAN-CSPGS Module

In target detection tasks where the object has an unconventional pose in a complex background, low-dimensional features such as shape and contour are crucial factors affecting detection performance. In order to fully extract and retain the low-dimensional features of the image without losing semantic information, this work introduces the CSPNet backbone network [18]. The CSPNet can enhance the learning ability of neural networks and promote the feature extraction capability of convolutional neural networks with less computational complexity.

Figure 5 illustrates the CSP structure, which divides the features into two parts. One part goes through dense convolutional blocks, and the other part undergoes regular convolution and directly connects to the output of the dense convolutional blocks. Transition layers are usually implemented with 1 x 1 convolutions or pooling operations.

Based on the CSP structure, the CSPNeXt-GS module is then designed to enhance the feature extraction capability of the convolutional neural network, as shown in Figure 6. By combining GSConv with the CSP structure, this module can strengthen the convolutional feature extraction while reducing computational complexity. Similarly, let $\sum$ represent the summation by elements, and the output can be written as

$$O = g_1^2(\text{Concat}(\sum(g_1^1(I), g_3^1(g_1^2(I))), g_1^2(I)))$$

(2)

In order to preserve deep features such as the unconventional pose, the branch D is replaced with the CSPNeXt-GS module in the structure of ELAN-GS, as shown in Figure 7.
This improvement enhances the subsequent convolutional layers and can extract deeper features while retaining the features of the A and B branches.

![Figure 7. ELAN-CSPGS structure.](image)

### 3.3. WIoU Loss

Most existing training works assume that the samples have high quality and focus on enhancing the fitting ability of the regression loss. However, blindly strengthening the bounding box regression of low-quality samples can hinder the generalization of the detector. To address this issue, the regression loss function WIoU is introduced in this work.

Let \((x_{gt}, y_{gt})\) and \((x, y)\) denote the coordinates of the center point of the object box and prediction box, respectively, and \(W_{gt}, H_{gt}\) be the width and height of the minimum enclosing box of the prediction box and the real box, as shown in Figure 8, and the loss can be expressed as

\[
L_{\text{loU}} = 1 - \frac{W_iH_i}{S_u}
\]  

(3)

where \(S_u\) denotes the intersecting part area and is calculated as

\[
S_u = w h + w_{gt}h_{gt} - W_iH_i
\]  

(4)

where \(w_{gt}\) and \(h_{gt}\) denote the width and height of the object box truth value, respectively; \(w\) and \(h\) denote the width and height of predicted object box, respectively; and \(W_i\) and \(H_i\) are the width and height of the intersection part of the real object box and predicted object box, respectively.

![Figure 8. Calculation of the WIoU loss function.](image)
Among the loss functions of WIoU, WIoUv3 [26] has better performance, and is illustrated as

\[ L_{WIoUv3} = r R_{WIoU} L_{IoU} \]  

\[ R_{WIoU} = \exp \left( \frac{(x - x_{gt})^2 + (y - y_{gt})^2}{(W^2_g + H^2_g)^*} \right) \]  

where \( r \) represents the focusing factor; \( \ast \) denotes separation from the computational map. When the prediction box overlaps with the object box, \( L_{IoU} \) will weaken the penalty for the geometric factor and focus attention on the centroid distance. Further, in the aggregation factor \( r \), an outlier \( \beta \) is introduced to assess the degree of anomaly of the anchor frame, and is denoted as

\[ \beta = \frac{L^*_{IoU}}{L_{IoU}} \in [0, +\infty) \]  

where \( L^*_{IoU} \) denotes the sliding average with momentum \( m \), and \( m \) is usually \( 1 - \sqrt[4]{0.05} \). \( t \) represents the training period, and \( n \) is the batch size. Therefore, the focusing factor \( r \) can be expressed as

\[ r = \frac{\beta}{\delta \alpha^{\beta - \delta}} \]  

where \( \alpha \) and \( \delta \) are hyper parameters, and the anchor frame will obtain the highest gradient gain when its outlier degree is some constant. The dynamic nature of \( L_{IoU} \) makes the quality classification criterion \( \beta \) for anchor frames dynamic as well, thus enabling WIoUv3 to dynamically formulate a gradient gain allocation strategy that best fits the current situation.

4. Experimental Results and Discussion

To validate the effectiveness of the proposed method, experiments and tests on human object detection were performed on the Ubuntu 18.04 64-bit operating system using the PyTorch based on Python 3.10. The training dataset is from https://aistudio.baidu.com/datasetdetail/229698 (accessed on 13 July 2023) and the COCO dataset [27], which mainly focus on the human target. The training and testing datasets in this work were randomly divided with a ratio of 8:2. Notably, all the models used below employ the learning rate 0.01, and take the training processing using this dataset, and were run on this platform consisting of an Intel® Xeon® CPU E5-2620 v4 2.1 GHz processor, 64 GB RAM, and an NVIDIA GeForce RTX 4090 24 GB GPU.

In our method, the anchor frames of the human object dataset are clustered by the use of a K-means++ clustering algorithm, and the prior bounding box size information obtained is shown in Table 1. The hyperparameters used for network training are set as follows: initial learning rate of 0.01, momentum of 0.937, and decay rate of 0.0005. The batch size is 16, and the Adam optimizer is used during training. The image size for training is set to 640 × 640 pixels, and the training period is 200 epochs.

<table>
<thead>
<tr>
<th>Feature Map Size</th>
<th>Initial Anchor Frame Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>80 × 80</td>
<td>(57, 52); (94, 150); (95, 452)</td>
</tr>
<tr>
<td>40 × 40</td>
<td>(233, 50); (271, 212); (281, 553)</td>
</tr>
<tr>
<td>20 × 20</td>
<td>(512, 85); (637, 636); (651, 252)</td>
</tr>
</tbody>
</table>

The quantitative evaluation metric for the experimental results is the average precision (AP) [28] at an intersection over union (IoU) threshold of 0.5. The calculation of AP is as follows
\[
AP = \frac{1}{m} \sum_{i} P_i = \int P(R) dR \\
mAP = \frac{1}{C} \sum_{i} AP_i
\]

where \( m \) represents the number of positive samples; \( C \) denotes the number of classes; \( P \) represents the maximum precision for a single sample, and \( AP \) represents the average precision for a class of sample.

### 4.1. Experimental Results

Figure 9 gives the values of the two loss functions with respect to the iteration, named CIoU [29] and WIoU [24]. As the iteration increases, both WIoU and CIoU eventually reach a converged state. Nevertheless, WIoU has a smaller loss value and converges faster compared to CIoU, thus demonstrating the effectiveness of the WIoU loss function in our method.

![Figure 9. Loss function iteration comparison.](image)

Figure 10 shows the P–R curves which intuitively display the recall and precision performance of the detector. The improved network has a larger area that indicates the better performance of the detector. Therefore, the improved network model can achieve better results.

![Figure 10. Comparison of network model P–R curves before and after improvement.](image)

To provide a visual demonstration of the detection performance of the proposed method on human objects, a comprehensive comparison is made with some existing classical deep learning models in terms of detection accuracy, spatial complexity, and
temporal complexity. Figure 11 illustrates five real-world images that were selected from transmission line inspection.

![Real-world images in transmission line inspection](image1)

**Figure 11.** Real-world images in transmission line inspection.

Figures 12–16 illustrate the results for human object detection with the comparison methods, including Faster R-CNN [13], YOLOv5 [30], YOLOv7-tiny, and YOLOv7 [25]. For the human objects, the improved network can detect all of them, while other networks show varying degrees of missed detection. For the small target in image3, the proposed method and YOLOv7 can detect it and outperform the other models. Moreover, the improved network can accurately box the human targets, with more precise detection box positions compared to the original Faster R-CNN, as seen in image1 and image4. Table 2 lists the accuracy of the human object detection. The proposed method has a lower false detection rate and leak detection rate. This demonstrates that the proposed improvements on Faster R-CNN achieve a better performance on the detection accuracy of human targets. It effectively reduces the rates of missed detection and false detection, improves the detection rate of small targets, and generates more accurate and effective detection boxes.

![Detection result of Faster R-CNN](image2)

**Figure 12.** Detection result of Faster R-CNN.

![Detection result of YOLOv5](image3)

**Figure 13.** Detection result of YOLOv5.

![Detection result of YOLOv7-tiny](image4)

**Figure 14.** Detection result of YOLOv7-tiny.

![Detection result of YOLOv7](image5)

**Figure 15.** Detection result of YOLOv7.
Table 2. Accuracy of human object detection.

<table>
<thead>
<tr>
<th>Model</th>
<th>Faster R-CNN</th>
<th>YOLOv5</th>
<th>YOLOv7</th>
<th>YOLOv7-Tiny</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>False detection rate</td>
<td>10.23%</td>
<td>5.47%</td>
<td>4.46%</td>
<td>5.76%</td>
<td>3.54%</td>
</tr>
<tr>
<td>Leak detection rate</td>
<td>19.30%</td>
<td>30.7%</td>
<td>26.31%</td>
<td>28.76%</td>
<td>6.21%</td>
</tr>
</tbody>
</table>

Table 3 presents a comparison of the time complexity among the mentioned methods. The YOLOv7 method is the least time-consuming, while the Faster R-CNN methods, being two-stage algorithms, have a higher time consumption. The proposed method is partially enhanced by the improvements, resulting in a high time complexity compared to the YOLOv7-tiny algorithms. However, it still has a lower time complexity compared to the YOLOv5 and YOLOv7.

Table 3. Evaluation of time complex.

<table>
<thead>
<tr>
<th>Model</th>
<th>Faster R-CNN</th>
<th>YOLOv5</th>
<th>YOLOv7</th>
<th>YOLOv7-Tiny</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total time</td>
<td>45.8 ms</td>
<td>14.0 ms</td>
<td>10.3 ms</td>
<td>6.3 ms</td>
<td>8.2 ms</td>
</tr>
</tbody>
</table>

Table 4 lists the maximum batch size (B) and memory usage (Memory, M) when training with different algorithms. The proposed algorithm achieves the highest accuracy and strikes a good balance between accuracy, speed, and model size, as seen in Table 4.

Table 4. Complexity comparison results of the different methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>B</th>
<th>M</th>
<th>FPS</th>
<th>Params</th>
<th>Flops</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLOv5</td>
<td>16</td>
<td>6378 M</td>
<td>119</td>
<td>14.5 M</td>
<td>15.8 G</td>
<td>83.4%</td>
</tr>
<tr>
<td>YOLOv7</td>
<td>16</td>
<td>5690 M</td>
<td>110</td>
<td>18.4 M</td>
<td>20.4 G</td>
<td>85.4%</td>
</tr>
<tr>
<td>YOLOv7-tiny</td>
<td>16</td>
<td>5600 M</td>
<td>117</td>
<td>12.4 M</td>
<td>13.1 G</td>
<td>85.2%</td>
</tr>
<tr>
<td>Ours</td>
<td>16</td>
<td>5836 M</td>
<td>117</td>
<td>9.7 M</td>
<td>10.4 G</td>
<td>92.6%</td>
</tr>
</tbody>
</table>

4.2. Absence of Ablation

To further demonstrate the performance of each improvement in the proposed method, the absence of ablation experiments are carried out. Table 5 gives the results of five cases, corresponding to the original YOLOv7-tiny, and its improvement with the ELAN_CSPGS, WIoU and K-means++, respectively. From Table 5, it can be observed that the above modules have gradually improved the detection accuracy of the model, while simultaneously reducing the model’s parameters and computational complexity, while maintaining a high frame rate.

Table 5. Ablation comparison.

<table>
<thead>
<tr>
<th>Test</th>
<th>FPS</th>
<th>Flops</th>
<th>Params</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>117</td>
<td>13.1 G</td>
<td>12.4 M</td>
<td>85.2%</td>
</tr>
<tr>
<td>Case 2</td>
<td>109</td>
<td>11.4 G</td>
<td>10.7 M</td>
<td>89.4%</td>
</tr>
<tr>
<td>Case 3</td>
<td>106</td>
<td>10.4 G</td>
<td>9.7 M</td>
<td>91.6%</td>
</tr>
<tr>
<td>Case 4</td>
<td>111</td>
<td>10.4 G</td>
<td>9.7 M</td>
<td>92.4%</td>
</tr>
<tr>
<td>Case 5</td>
<td>117</td>
<td>10.4 G</td>
<td>9.7 M</td>
<td>92.6%</td>
</tr>
</tbody>
</table>
4.3. Further Discussion

Recently, the latest version YOLO series methods have appeared [31,32]. Figures 17 and 18 illustrate the results of human object detection. It can be seen that YOLOv8 and YOLOv9 can obtain the desired results. The performance of the training dataset is shown in Table 6. The confidence is relatively low due to the fact that the YOLOv8n.pt and YOLOv9-c.pt are used for the pre-training model, respectively. Nevertheless, the proposed model has the advantage of less parameters and an acceptable accuracy in transmission line inspection, making it suitable for running on mobile or device endpoints.

![Figure 17. Detection result of YOLOv8.](image1)

![Figure 18. Detection result of YOLOv9.](image2)

Table 6. Performance of the latest version YOLO series.

<table>
<thead>
<tr>
<th>Model</th>
<th>FPS</th>
<th>Params</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLOv8</td>
<td>257</td>
<td>11.6 M</td>
<td>91.4%</td>
</tr>
<tr>
<td>YOLOv9</td>
<td>178</td>
<td>25.3 M</td>
<td>92.9%</td>
</tr>
</tbody>
</table>

5. Conclusions

This paper proposes an enhanced single-stage neural network based on the YOLO-tiny model for human object detection in transmission line inspection. The model incorporates a lighter GSCConv module to enhance the original ELAN-tiny module and combines it with the CSP structure to make the network less sensitive to low-contrast targets. The WIoU loss function is utilized to replace the original loss function, further enhancing the detection performance. The experimental results demonstrate that the proposed network model achieves a higher detection accuracy compared to some existing and commonly used models and has fewer parameters and a lesser computational complexity. In the near future, the proposed method will be applied to endpoint devices for transmission line inspection.

Author Contributions: Methodology, J.T.; Software, J.N.; Validation, Z.C. and Z.H.; Formal analysis, X.X.; Writing—original draft, C.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the “Research on Safety Risk Identification and Operation Control Technology for Power Grid Engineering Based on 3D Spatial Information Fusion” project of the State Grid Corporation of China (Grant No. 5700-202233182A-1-1-ZN) and the APC was funded by the same project.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: Authors Jianglong Nie, Zhao Chen and Zhouqiang He were employed by the State Grid Gansu Electric Power Company. The remaining authors declare that the research was
Conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.