


Article

MGFNet: A Progressive Multi-Granularity Learning Strategy-Based Insulator Defect Recognition Algorithm for UAV Images

Zhouxian Lu ¹, Yong Li ^{1,2,*}  and Feng Shuang ¹

¹ Guangxi Key Laboratory of Intelligent Control and Maintenance of Power Equipment, School of Electrical Engineering, Guangxi University, Nanning 530004, China; 2012391028@st.gxu.edu.cn (Z.L.); fshuang@gxu.edu.cn (F.S.)

² Hubei Key Laboratory of Intelligent Robot, Wuhan Institute of Technology, Wuhan 430205, China

* Correspondence: yongli@gxu.edu.cn

Abstract: Due to the low efficiency and safety of a manual insulator inspection, research on intelligent insulator inspections has gained wide attention. However, most existing defect recognition methods extract abstract features of the entire image directly by convolutional neural networks (CNNs), which lack multi-granularity feature information, rendering the network insensitive to small defects. To address this problem, we propose a multi-granularity fusion network (MGFNet) to diagnose the health status of the insulator. An MGFNet includes a traversal clipping module (TC), progressive multi-granularity learning strategy (PMGL), and region relationship attention module (RRA). A TC effectively resolves the issue of distortion in insulator images and can provide a more detailed diagnosis for the local areas of insulators. A PMGL acquires the multi-granularity features of insulators and combines them to produce more resilient features. An RRA utilizes non-local interactions to better learn the difference between normal features and defect features. To eliminate the interference of the UAV images' background, an MGFNet can be flexibly combined with object detection algorithms to form a two-stage object detection algorithm, which can accurately identify insulator defects in UAV images. The experimental results show that an MGFNet achieves 91.27% accuracy, outperforming other advanced methods. Furthermore, the successful deployment on a drone platform has enabled the real-time diagnosis of insulators, further confirming the practical applications value of an MGFNet.

Keywords: drone insulator inspection; defect detection; deep learning; convolutional neural network



Citation: Lu, Z.; Li, Y.; Shuang, F. MGFNet: A Progressive Multi-Granularity Learning Strategy-Based Insulator Defect Recognition Algorithm for UAV Images. *Drones* **2023**, *7*, 333. <https://doi.org/10.3390/drones7050333>

Academic Editor: Pablo Rodríguez-González

Received: 30 March 2023

Revised: 12 May 2023

Accepted: 15 May 2023

Published: 22 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Insulators play a crucial role in power transmission lines, providing both mechanical support and electrical insulation for equipment and conductors. By preventing the current flow to the ground or other conductors, they ensure safe operation and prevent safety accidents such as fires and explosions. However, insulators are easy to be damaged by various factors, such as ultraviolet radiation, pollution, and lightning strikes. This damage can reduce the insulation level of power equipment, impacting the power system's stability and reliability. Manual inspection is currently the primary method used to inspect insulators, as showed in Figure 1a. However, this method is expensive, time-consuming, and inefficient, which increases the risk of system failures and safety incidents. To address these issues, researchers are developing unmanned aerial vehicles (UAV)-based insulator defect recognition algorithms. As showed in Figure 1b, UAVs conduct aerial surveys of power equipment and utilize on-board algorithms for insulator localization and defect diagnosis in the aerial images. These algorithms have the potential to improve the efficiency and accuracy of insulator inspections, leading to more reliable and safe power systems.

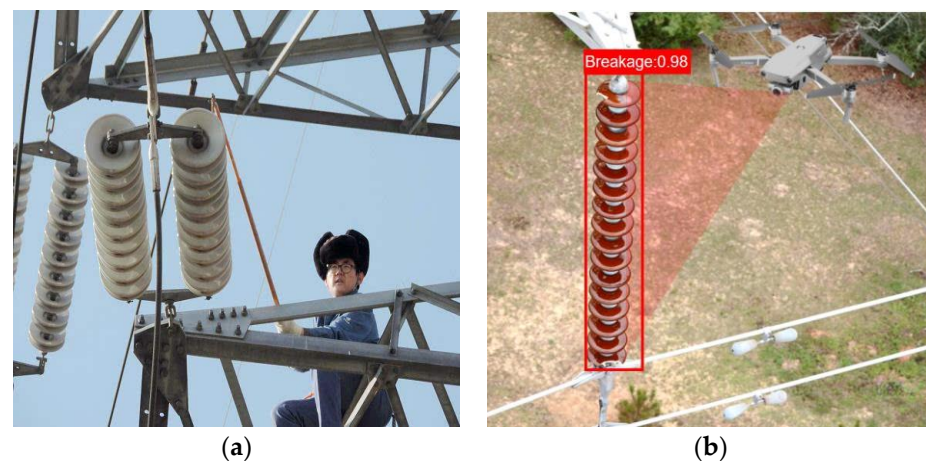


Figure 1. Some power line inspection scenes: (a) manual inspection; (b) UAV inspection.

Defect recognition methods can also be roughly categorized into weakly supervised defect recognition and strong supervised defect recognition. Weakly supervised learning methods alleviate the problems of sparse defective samples and imbalanced data. Weakly supervised methods can be divided into two types: unsupervised learning and few-shot learning. Unsupervised methods train models without defect samples, which addresses the challenge of collecting sufficient defect samples. This kind of method often relies on autoencoders or generative adversarial networks and can automatically learn features for defect detection. For example, reference [1] proposed an unsupervised learning-based approach for detecting defects in catenary rod-insulators. The proposed method consisted of three stages: separating insulator regions, reconstructing and recognizing insulator pieces using a convolutional autoencoder network, and evaluating the defect levels using a clustering algorithm. The DefGAN [2] was a novel approach for automating the detection of defects in high-speed railway catenary insulators. By using a pitted latent representation to generate defective samples, the DefGAN improved the reliability of the defect detection classifier. Few-shot learning requires transferring prior knowledge from the source domain to the target domain, and then using contrastive learning and the support set created by a small amount of data to accomplish the classification task. Reference [3] proposed a few-shot defect recognition (FSDR) for real industrial scenarios with insufficient training samples. The proposed method achieves defect recognition by a coarse-to-fine manner with a dynamic weighting and joint metric.

Although unsupervised methods solve the problem of scarce samples in insulation defect detection, their classification accuracy is often affected by data quality, resulting in poor robustness and difficulty in meeting the expected accuracy requirements. With the accumulation of defect insulator image samples in recent years, the training requirements for the strong supervised methods can now be met. Its detection accuracy is better than unsupervised methods and can better meet the requirements of intelligent inspection. For example, Ref. [4] proposed a novel method called a Box-Point Detector for the fault diagnosis of insulators in aerial images. Ref. [5] proposed an intelligent fault detection method for overhead line insulators based on aerial images and an improved YOLOv3 [6], which uses a densely connected feature pyramid network (FPN) to improve detection performance and reduce the risk of network overfitting. The YOLOv3-DenseNet algorithm [7] enhances feature reuse and fusion by utilizing dense blocks, which effectively improves defect detection accuracy in printed circuit boards (PCBs). However, the methods [4,5,7] mentioned above all use single-granularity feature extraction, which makes it difficult to capture the defect feature information in different granularities. This can result in the missing detection of very small defects and the false detection of very large defects. In addition, single-granularity feature extraction will result in redundant feature calculations, increasing the computational complexity of the network.

Although these strong supervised methods have shown some success, their reliance on single-granularity feature extraction renders them incapable of capturing the multi-scale variations present in insulator defects, as illustrated in Figure 2. Consequently, the current approaches are prone to overlooking very small defects and misidentifying large ones.

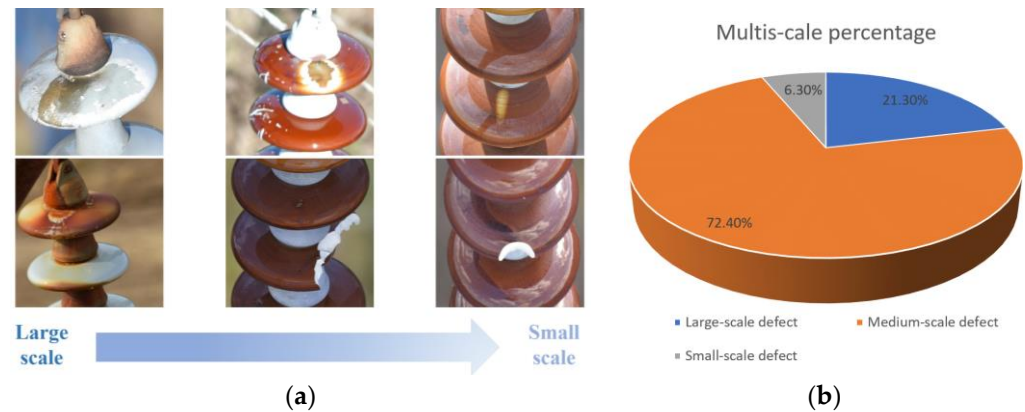


Figure 2. Multiscale variation of insulator defects. (a) Samples from large-scale defects to small-scale defects; (b) the percentage of multiple-scale defects in the data set.

In summary, the current weakly supervised methods in the field of defect recognition suffer from low accuracy and poor robustness, as they are easily affected by environmental and lighting factors. Although strong supervised methods can achieve better robustness, they are only suitable for recognizing obvious defects and have limitations in accurately identifying tiny defects and extracting the subtle features. In addition, many insulators are rectangles with a large aspect ratio, while the input of deep learning models requires resizing the insulator image into a square with an aspect ratio of 1. This will cause the severe distortion of the already small insulator. As shown in the Figure 3, image processing can make less obvious defects even less apparent. To address these issues, this paper proposes a multi-granularity fusion network (MGFNet) to diagnose the health status of the insulator. An MGFNet includes a traversal clipping module (TC), progressive multi-granularity learning strategy (PMGL), and region relationship attention module (RRA).

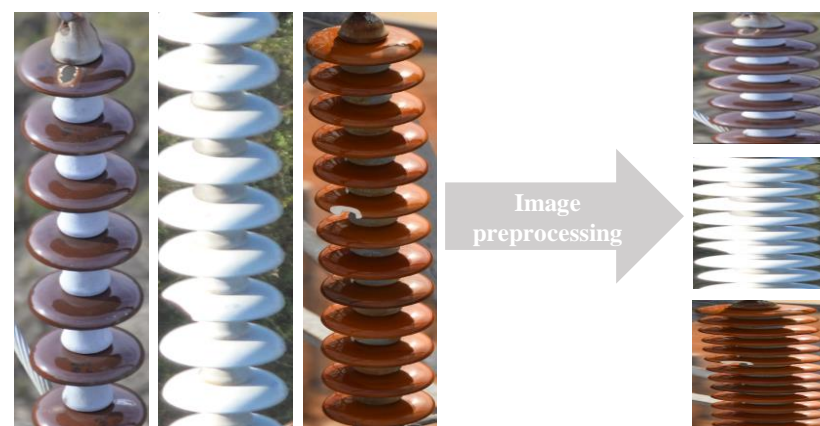


Figure 3. Image preprocessing results in serious distortion of insulator defects.

The contributions of this paper can be summarized as follows:

- To solve the distortion issue caused by images pre-processing, we propose a novel traversal clipping module (TC). A TC can divide insulators into multiple patches according to their aspect ratios and traverse each patch for diagnosis. A TC not only mitigates image distortion but also increases the number of data samples, playing a role of data enhancement.

- We propose a novel progressive multi-granularity learning strategy (PMGL) that leverages convolution operations at various granularities to extract the feature information of different granularities in images, including detailed information at low levels and semantic information at high levels. This strategy enables the network to achieve a good recognition performance for defects in different granularities. Moreover, we utilize KL divergence to guide multi-granularity features to focus on different objectives and extract complementary information.
- To improve the ability to distinguish between defect and normal regions, we propose a region relation attention module (RRA) that performs a non-local interaction between local features. RRA aggregates and adjusts non-local information in the feature map, which helps the model to better understand the relationships between normal and defective regions in the image, thereby improving its performance in visual analysis and recognition.
- Based on the above three points, we propose a multi-granularity fusion defect network (MGFNet) for insulator defect recognition. The experiments show that an MGFNet achieves 91.27% accuracy, outperforming advanced methods, with a parameter size of 84.1 megabytes and a speed of 126.2 images/s, demonstrating its practical value.

This paper is organized as follows. Section 2. Related Work summarizes the relevant papers. Section 3. The Proposed MGFNet presents the details of the proposed network. Section 4. Experimental Results and Analysis shows the experiment results and related analysis. Finally, Section 5. Conclusion presents the summary and future work.

2. Related Work

This section mainly discusses the public networks for defect detection and attention mechanisms. We summarized the related works from these two aspects.

2.1. Defect Detection

Recently, there has been rapid development in the defect detection methods for surfaces, which can be categorized into four types: conventional statistical, spectral, model-based, and emerging machine learning. Statistical approaches typically use pixel distribution and variation patterns to evaluate the defect areas. For instance, Zhao et al. [8] utilized superpixels to group pixels with similar visual properties, thereby aggregating defective areas into a superpixel. However, statistical methods are susceptible to interference from illumination variations or pseudo defect visits. Spectral approaches, on the other hand, aim to find a special transform domain where defect objects can be more easily and completely separated from both local and global backgrounds. For example, Sharifzadeh et al. [9] employed the Hough transform [10] to detect holes, scratches, coil breaks, and rust defects on cold-rolled steel strips. Nonetheless, the spectral method faces difficulties in representing miscellaneous defects and stochastic background variations on textured surfaces. Model-based methods tend to map the image to a low-dimensional feature space to filter out noise and obtain better feature representation. For example, Yang et al. [11] proposed an active contour model (ACM)-based defect detection method to effectively segment defect features from a complex background. However, the ACM-based method struggles with calculating the convergence position due to the lack of constraints.

In recent years, there has been rapid development in the application of deep learning techniques for defect detection. Deep learning-based methods can be broadly categorized into three types: supervised learning, unsupervised learning, and weakly supervised learning. The objective of supervised learning method [12] is to model a conditional distribution between input vectors (surface images) and target vectors. However, supervised learning heavily relies on large amounts of training data and may lead to severe overfitting when trained on small datasets. Unsupervised learning aims to recover the original data from the abstracted data with minimal loss, and it learns the hidden data features. For instance, P. Perera et al. [13] utilized deep convolutional generative adversarial networks (GAN) to detect defects on textured surfaces, requiring only positive samples without any defect

samples or manual labels. Nonetheless, unsupervised learning is sensitive to noise and is heavily influenced by initial values. Few-shot learning (FSL) is a typical example of weakly supervised learning methods. FSL aims to mimic human learning abilities, requiring only a few samples to complete the learning and possessing strong generalization capabilities. FSL transfers extensive prior knowledge from the source domain to the target domain, then compares the feature similarity between the support set and the query set for classification.

2.2. Attention Mechanism

The attention mechanism can effectively focus on discriminative regions and filter out redundant information. For example, Hu et al. [14] designed the squeeze-and-excitation module (SENet) to obtain the weights for each channel and suppress or enhance channels to improve recognition accuracy. Wang et al. [15] improved upon the SENet [14] by proposing the efficient channel attention (ECA), which aggregates cross-channel information through a one-dimensional convolutional layer to obtain more accurate attentional information. CBAM [16] considers that not only do channels contain rich attention information, but also the interpixel information on the map has significant attention information. CBAM builds two submodules (spatial attention module and channel attention module) to aggregate attention information from both spatial and channel aspects, obtaining more comprehensive and reliable attention information. The SK-Net [17] argues that different input images require different receptive fields. The SK-Net designs three parts (split, fuse, and select) that enable each neuron to adaptively adjust its receptive field size according to the scale of the input information. Although the defects on the insulator are very small, the difference between the normal and defective areas is obvious, and the features of the normal and defective regions are stable.

3. The Proposed MGFNet

3.1. MGFNet Overview

The network architecture of the MGFNet is illustrated in Figure 4, which consists of four main parts: the backbone network, the traversable clipping (TC) module, the progressive multi-grained learning strategy (PMGL), and the region relation attention (RRA) module. The MGFNet uses ResNet50 as its backbone network, which contains four residual blocks: Res_Block0, Res_Block1, Res_Block2, and Res_Block3. As depicted in Figure 5, the TC first divides the insulator images into multiple patches according to their aspect ratios, avoiding distortions caused by defects. Then, multiple local patches are input into the MGFNet for traversable diagnosis. The PMGL employs four steps to learn multi-grained features and integrates them to obtain a more comprehensive feature representation. The RRA module learns the correlation between non-local features, enabling the network to better distinguish the normal and abnormal regions. Each module will be elaborated in detail below.

3.2. Traversal Clipping Module

Due to the structural characteristics of insulators, image pre-processing often causes the serious distortion of insulators, making it more difficult to identify. The reason for the distortion is that the network requires the input image to be square (i.e., have an aspect ratio of 1), while insulator images are typically rectangular with a large aspect ratio. Resizing the insulator image forcibly to a square will cause significant distortion. Therefore, our approach is to crop the insulator image into multiple patches with aspect ratios close to 1, and then input these patches into the network for recognition in sequence. Based on the above analysis, we propose the TC. As shown in the Figure 6, the TC divides the complete insulator image into n insulator patches according to the aspect ratio $r = h/w$ (where h and w represent the height and width of the input image, respectively). The TC can be represented as follows:

$$x_1, x_2, \dots, x_n = TC(x, k) \quad (1)$$

where k ($k > 1$) is a hyperparameter used to control the aspect ratio of patches to approach k . The number of patches output by the TC is determined by the following formula:

$$n = \lceil r/k \rceil \tag{2}$$

where $\lceil \cdot \rceil$ is the quotient of r divided by k . k has a great effect on the distortion of the image and the speed of the model. For example, as k increases, the aspect ratio of patches increases, and distortion also increases accordingly, but the number of outputs decreases, which speeds up the recognition of an image. Specifically, the choice of k will be discussed in the experimental section.

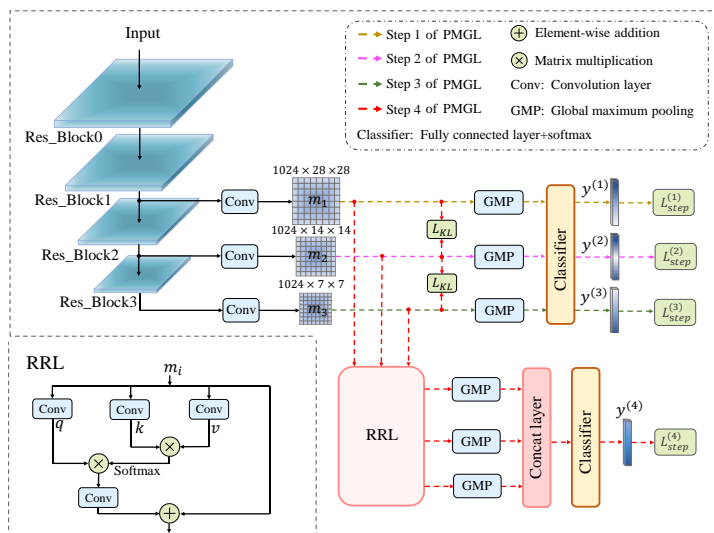


Figure 4. The structure of MGFNet. MGFNet mainly adopts MGFNet to use four steps to learn multi-granularity features. Moreover, RRA is adopted to learn the correlation between non-local features, enabling the network to distinguish better between normal and abnormal areas.

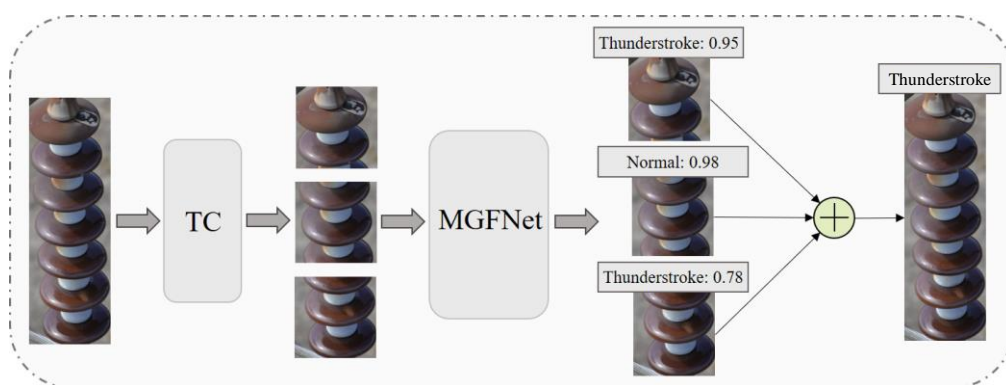


Figure 5. The defect recognition stage uses the proposed MGFNet to diagnose the health status of the insulator.

3.3. Progressive Multi-Granularity Learning Strategy

Due to the impact of different environments (such as lightning and acid rain) on insulators, the discriminative parts of the defects are multi-granularity and irregular. Therefore, obtaining the multi-granularity features of the defects is crucial for the performance of identification. To obtain richer multi-granularity features, we adopt the PMGL to address this issue. As shown in the Figure 4, the PMGL is divided into four steps (represented by yellow, purple, green, and red arrows for steps 1–4) to learn the multi-granularity information. Since step 4 is different from step 1–3, we introduce steps 1–3 first. In step 1 of the PMGL, the Res_Block0-1 shallow network is first trained to obtain the coarse-grained

feature m_1 . In step 2 of the PMGL, the deeper network layer Res_Block0-2 with a larger receptive field is gradually trained to obtain the medium-grained feature m_2 . In step 3 of the PMGL, the deep network Res_Block0-3 is trained to cover the entire image and obtain the fine-grained feature m_3 . At this point, the PMGL completes a feature extraction from coarse-grained to fine-grained. Specifically, in the process of step i ($i < 3$), a patch x_n of the insulator image x is input into Res_Block0- i to obtain a feature m_i , which is fed to a global maximum pooling layer and a fully connected layer to get prediction result y^i . Then, we use a cross entropy loss L_{CE} to update the parameters of the network in each step by back propagation. It is important to note that all parameters are optimized in the current step, regardless of whether they were updated in a previous step. The loss function of step i $L_{pro}^{(i)}$ can be expressed as follows

$$L_{step}^{(i)} = L_{CE}^{(i)}(y_j^{(i)}) = -\sum_{x_j \in D} y_j^{(i)} \ln y_j^{(i)} \tag{3}$$

where D is the training set and x_j is the j -th image in D . After step 1–3, the MGFNet has enabled the network to learn features at multi-granularities (i.e., m_1, m_2, m_3). However, simply using the MGFNet will not result in diverse features and lacks non-local information interaction, as the multi-granularities information obtained through the MGFNet may be concentrated in similar areas. To address this problem, in step 4, the KL divergence and RRA (introduced in Section 3.5) is designed to guide multi-granularities features m_1, m_2 , and m_3 to focus on different regions, increasing the probability of capturing less obvious defective areas. Specifically, as shown by the red arrows in the Figure 4, the multi-granularity features (m_1, m_2 , and m_3) obtained in steps 1–3 are inputted into L_{KL} , and by maximizing the KL divergence between the features from different steps, we force multi-granularity features to learn different features. The L_{KL} calculation process is as follows

$$L_{KL}(m_i, m_j) = -\sum_{i=1}^3 \sum_{j=3-i}^3 m_i \log\left(\frac{m_i}{m_j}\right) \tag{4}$$

where m_i and m_j are the multi-granularity features from different steps.

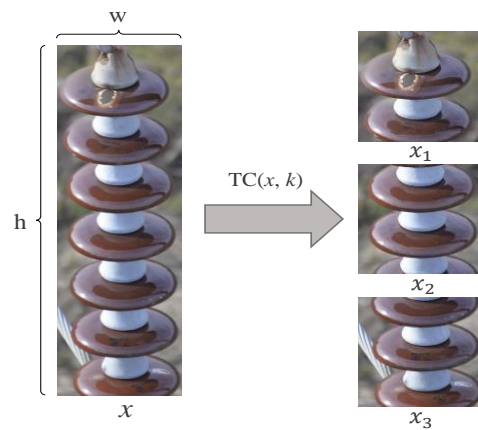


Figure 6. TC divides the complete insulator image into n insulator patches according to the aspect ratio.

3.4. Local Relationship Attention Module

Compared to typical recognition tasks, identifying insulator defects poses challenges due to the absence of fixed semantic information and the wide variety of visual features exhibited by defects. However, there is a discernible contrast between the defect and normal regions. Therefore, understanding the differences between features in the various regions can enhance the ability to distinguish between defect and normal features. Unfortunately, many current methods extract abstract features directly from the entire image, neglecting

the relationships between local features. To overcome this limitation, we introduce RRA, which captures the relationships between non-local local features. The RRA structure is shown in Figure 4. In step 4, m_1, m_2, m_3 are not only optimized by the KL divergence but also input into the RRA for non-local features interaction to the obtained enhanced features \hat{m}_i . The RRA process is as follows

$$\begin{aligned} q, k, v &= \text{Conv}(m_i) \\ a_{q,k} &= \text{Softmax}(qk^T / \sqrt{d_k}) \\ \hat{m}_i &= a_{q,k}v \end{aligned} \quad (5)$$

where $a_{q,k}$ means the similarity between q and k . Therefore, we obtained an enhanced feature \hat{m}_i . Then, the enhanced features \hat{m}_1, \hat{m}_2 , and \hat{m}_3 are concatenated to obtain a more comprehensive feature \hat{m}_{cat} , and the prediction result $y^{(4)}$ is obtained through a classifier. The calculation process is as:

$$y^{(4)} = \text{Fc}(\text{GMP}(\text{Concat}(\hat{m}_1, \hat{m}_2, \hat{m}_3))) \quad (6)$$

Finally, the cross-entropy loss function is used to calculate the loss between $y^{(4)}$ and the label, and the calculation formula is as follows:

$$L_{CE}^{(4)} = - \sum_{x_j \in D} y_j^{(4)} \ln y_j^{(4)} \quad (7)$$

Step 4 of the PMGL consists of the loss function $L_{CE}^{(4)}$ and L_{KL} (as introduced in Section 4.2.3). Thus, the loss function of Step 4 in the PMGL can be described as follows:

$$L_{step}^{(4)} = \alpha L_{CE}^{(4)} + \beta L_{KL} \quad (8)$$

where α and β are the equilibrium parameters of the loss function. The parts of the MGFNet are introduced, and its detailed process is shown in Algorithm 1.

Algorithm 1: The training process of MGFNet.

Input: training set D , model parameter θ , hyperparameter k, α, β

while $n \leq N$ **do**

randomly sample x **in** D

$[x_1, x_2, \dots, x_n] = \text{TC}(x, k)$

for x_i **in** $[x_1, x_2, \dots, x_n]$:

for i **in** $\text{rage}(4)$: ## the 4 steps of MGFNet

if $i < 3$:

$m_i = \text{Res_Block}_{0-i}(x_i)$

$y^{(i)} = \text{Fc}(\text{GMP}(m_i))$

$\theta_{0-i} = \theta_{0-i} - \nabla_{\theta_{0-i}} L_{step}^{(i)}(y^{(i)})$ ## θ_{0-i} is the parameter of Res_Block0- i

else:

$\hat{m}_1, \hat{m}_2, \hat{m}_3 = \text{RRA}(m_1, m_2, m_3)$

$y^{(4)} = \text{Fc}(\text{GMP}(\text{Concat}(\hat{m}_1, \hat{m}_2, \hat{m}_3)))$

$L_{step}^{(4)} = \alpha L_{CE}^{(4)}(y^{(4)}) + \beta L_{KL}(m_1, m_2, m_3)$

$\theta_{0-3} = \theta_{0-3} - \nabla_{\theta_{0-3}} L_{step}^{(4)}$

Return model parameter θ

3.5. MGFNet-Based Two-Stage Insulator Defect Detection Algorithm

During the training of the MGFNet, we used insulator images artificially extracted from UAV images without complex background interference. However, actual UAV images contain complex background interference, and the MGFNet is unable to effectively detect insulator defects in aerial images. Therefore, in this section, we propose a two-stage insulator defect detection algorithm based on the MGFNet for the accurate identification of

insulator defects in UAV aerial images. As shown in Figure 7, the MGFNet-based two-stage insulator defect detection algorithm consists of two stages: an insulator extraction stage and defect recognition stage. The insulator extraction stage uses objective detection models (such as the Faster RCNN, SSD [18], YoLo) to locate and extract the insulator in aerial images to eliminate the influence of complex backgrounds. The defect recognition stage uses the proposed MGFNet to diagnose the health status of the insulator. Compared to the one-stage detection model, the two-stage insulator defect detection algorithm often has a higher detection accuracy and better robustness due to the elimination of complex background interference and the narrowing of the recognition scope. Furthermore, this algorithm offers excellent flexibility, enabling the use of various objective detection models to meet the specific requirements of different platforms. For instance, YoLov5 can be applied to unmanned aerial vehicle platforms with limited computing power, while Faster RCNN is better suited for platforms that emphasize accuracy and possess a superior computing power for object detection.

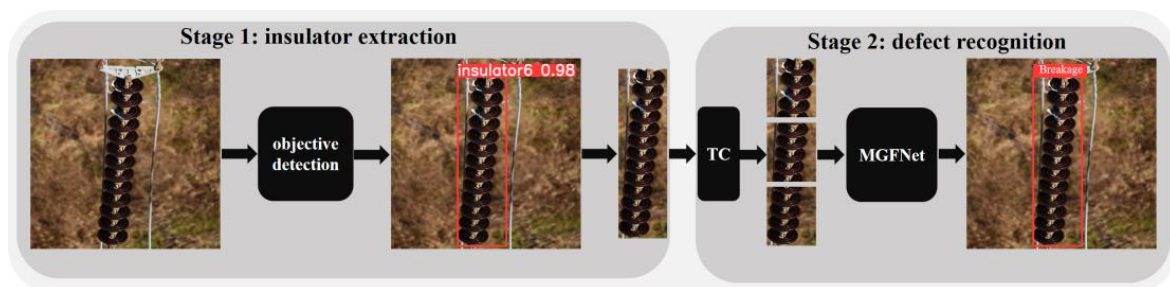


Figure 7. The structure of the MGFNet-based two-stage insulator defect detection algorithm.

4. Experimental Results and Analysis

In this section, we present a detailed description of the experimental setup and extensively evaluate the effectiveness of our proposed methods and modules through numerous experiments.

4.1. Implementation Details

4.1.1. Training Process

In this paper, the experimental environment and parameter configuration are shown in Table 1. All experiments are conducted on the PyTorch platform and a single GPU (NVIDIA TITAN V). The insulator images are inputted into the TC for cropping to obtain local patches of insulators, then the patches are resized to 224×224 and input to the PMLNet. The training of the PMLNet employs a batch size of 8 for 50 epochs. During training, the stochastic gradient descent (SGD) optimizer is used with a momentum of 0.9 and a weight decay of 0.0005. The learning rate is set to 0.001 for the first 20 epochs and is multiplied by 0.9 every 2 epochs thereafter. This way, the learning rate is gradually reduced to better train the network and improve its performance. The hyperparameters k , α and β are set to 1.3, 0.8, and 0.2, respectively.

4.1.2. Dataset Acquisition

In this paper, the insulator dataset is collected by DJI M300RTK UAV to take aerial images of insulators at different places and at different time. The DJI M300RTK UAV, as depicted in Figure 8, boasts numerous advantages, including a long endurance of 55 min and six-direction positioning obstacle avoidance function. We equipped the UAV with the Zenmuse P1 full-frame camera and deployed it to patrol power lines and capture images of insulators. In addition, we also collected a small number of defective insulator images from the internet as a supplement.

The insulator dataset is divided into four classes: normal insulators, thunderstroke insulators, breakage insulators, and pollution insulators, where thunderstroke, breakage and pollution insulators are collectively referred to as abnormal insulators. The number

of samples in each category of the dataset is shown in Figure 9. The training set contains 1316 images (including 155 normal insulators, 493 thunderstroke insulators, 503 breakage insulators, and 165 pollution insulators), and the test set contains 344 images (including 144 normal, 50 pollution, 78 breakage, and 72 thunderstroke insulators), as shown in the Figure 10a for each class. After the TC, the training set contains 3495 patches (including 1263 normal patches, 208 pollution patches, 916 breakage patches, and 1111 thunderstroke patches). It can be seen that the number of data samples is greatly increased after the TC. By comparison, it shows that the number of data samples has significantly increased after the application of the TC. Thus, the TC not only compels the network to recognize more subtle features but also acts as a data augmentation technique. Figure 10b shows the distribution of each class after the TC.

Table 1. Experimental environment and parameter configuration.

Hardware platform	CPU GPU Memory size	Intel® X®(R) Gold 6136 CPU @3.00 GHz TITAN v@12 GB 187 GB
Software platform	Operating system version Deep learning framework Python version	Ubuntu 16.04.6 LTS Pytorch 1.4.0 3.8.12
Hyperparameters	Batch-size	8
	Epoch	50
	Input-size	224 × 224
	Learning rate	0.001
	Optimizer	SGD
	Momentum coefficient	0.9
	Weight decay coefficient	5×10^{-4}
	k	1.3
	α	0.8
	β	0.2



Figure 8. The DJI M300RTK UAV.

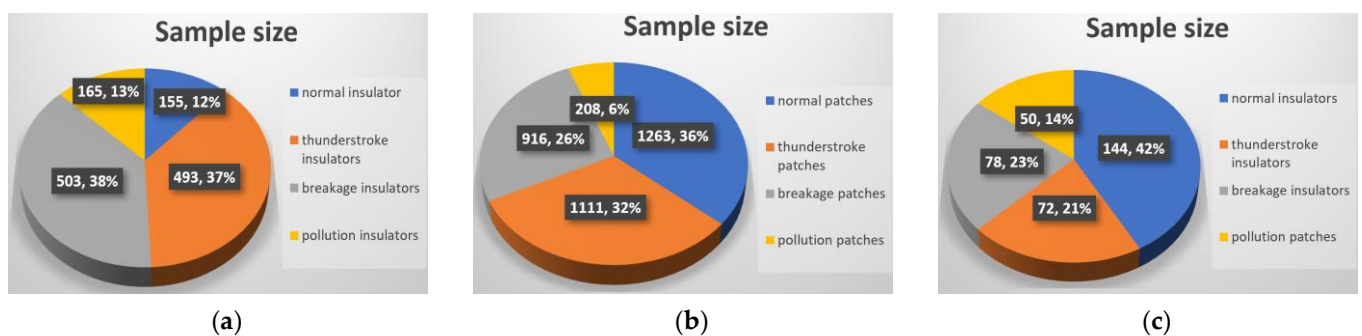


Figure 9. Number of samples in each category of the dataset. (a) Number of each category in the training set. (b) Number of each category in the training set after TC. (c) Number of each category in the test set.

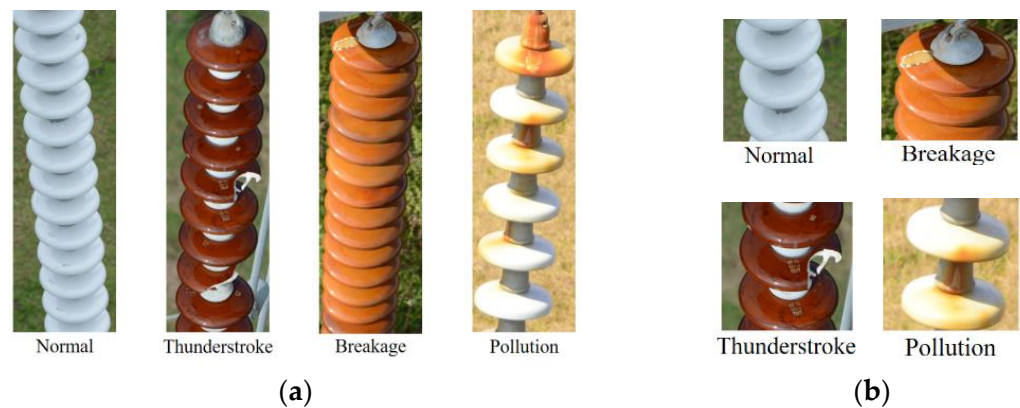


Figure 10. Samples of each category in the data set (a) Original images; (b) the image after TC.

4.1.3. Metrics

To verify the effectiveness of the proposed the MGFNet, we adopted three widely used metrics to quantitatively evaluate the performance of our defect recognition method, i.e., accuracy (*Acc*), Prams (megabytes), and Speed (images/s).

The accuracy (*Acc*) measures the proportion of correctly classified samples on the test data to the total number of samples. The specific calculation formula is as follows:

$$Acc = \frac{TP + TN}{TP + FP + FN + TN} \quad (9)$$

where *TP* (True Positives) represent the number of positive samples that are correctly identified as positive by the classifier; *FP* (False Positives) represent the number of negative samples that are incorrectly identified as positive; *TN* (True Negatives) represent the number of negative samples that are correctly identified as negative by the classifier; *FN* (False Negatives) represent the number of positive samples that are incorrectly identified as negative.

The Prams (megabytes) represent the number of parameters that need to be trained in the model.

Speed (images/s) refers to the number of images that a model can process per second.

4.2. Ablation Studies

To verify the effectiveness of the proposed module, we conduct multiple ablation experiments on the ES, PGL, and RRA. Four models are set up: Model (a) ResNet50, Model (b) ResNet50 with TC without PGL and RRA, Model (c) ResNet50 with TC and PGL but without RRA, and Model (d) ResNet50 with all modules (MGFNet). The results of the ablation experiments are presented in Table 2. Comparing Model (a) and Model (b), we observe significant improvements in accuracy for each class, particularly for normal, damaged, and lightning-struck classes, which are increased by 31.69%, 38.47%, and 52.94%, respectively. We note that Model (b) has a slightly lower speed than Model (a), but the speed is still fast enough to meet real-time requirements. After comparing Model (b) and Model (c), the results indicate that Model (c) has a 2.61% improvement in accuracy for all classes, with a significant 10.25% improvement in accuracy for the damaged class. This demonstrates that the proposed PGL can effectively alleviate the scale diversity problem of defects. Furthermore, comparing Model (c) and Model (d), an overall accuracy improvement of 0.58% is observed, with a 2.57% increase in accuracy for the damaged class. This proves that RRA can play an essential role in fine-grained defect recognition.

Table 2. Results of ablation studies. \checkmark means using module during training.

Model		(a)	(b)	(c)	(d)
ResNet50		\checkmark	\checkmark	\checkmark	\checkmark
TC			\checkmark	\checkmark	\checkmark
MGFNet				\checkmark	\checkmark
RRA					\checkmark
Acc (%)	Average	56.39	88.08	90.69	91.27
	Normal	70.13	95.12	97.50	98.31
	Anomaly	80.49	86.49	92.00	92.50
	Pollution	81.99	88.99	90.00	90.01
	Breakage	30.76	77.23	81.21	82.05
	Thunderstroke	38.89	85.12	87.83	87.99
Params (megabytes)		42.5	42.5	69.16	84.1
Speed (images/sec)		419.6	138.3	130.3	126.2

4.2.1. Effectiveness of Each Learning Stage and Multi-Stage Fusion

For the effectiveness of each learning stage and multi-stage fusion, we conducted a series of experiments on Model (c) with different learning stages. Table 3 demonstrates that the accuracy of each category improves gradually from 1 to 3 steps, due to the increased depth of the network which expands the receptive field of features and gathers more semantic information [19]. In addition, we also observe that multi-granularity fusion (without KL divergence) further improves the accuracy of the model, demonstrating that multi-granularity information fusion has a significant improvement effect. Furthermore, multi-granularity fusion (with KL divergence) achieved the best classification accuracy, which demonstrates that the features optimized by KL divergence have richer semantic information.

Table 3. The effectiveness studies of each learning stage and multi-stage fusion.

Model (c)	KL Divergence	Acc (%)					
		Average	Normal	Abnormal	Pollution	Breakage	Thunderstroke
Step 1 of PMGL	80.12	89.21	83.35	78.65	80.34	62.72	80.12
Step 2 of PMGL	85.35	93.36	89.88	82.36	85.15	71.62	85.35
Step 3 of PMGL	89.58	95.15	91.56	87.11	89.99	79.71	89.58
Multi-granularity fusion	90.38	97.28	91.83	88.52	81.09	87.93	90.38
Multi-granularity fusion	91.27	98.31	92.50	90.01	82.05	87.99	91.27

4.2.2. Visualization of the RRA

To better understand the proposed RRA, we use Grad-CAM [19] to visualize the RRA. In Grad-CAM, the color represents the gradient value calculated by the neural network for each pixel, which is then mapped to the input image and encoded into different colors. Red represents a positive gradient, blue represents a negative gradient, and yellow and green represent a neutral gradient. Strong positive gradient values indicate that the region contributes more to the prediction result, while negative gradient values indicate the opposite. Observing Grad-CAM can help us understand which areas the RRA focuses on. Figure 11 clearly illustrates that without the RRA, attention is predominantly directed towards normal areas while subtle defect areas receive relatively low attention weights. The RRA not only focuses on normal areas but also captures subtle defect areas well. In other words, our RRA has a larger receptive field through non-local learning, which strengthens the ability to extract fine-grained features.

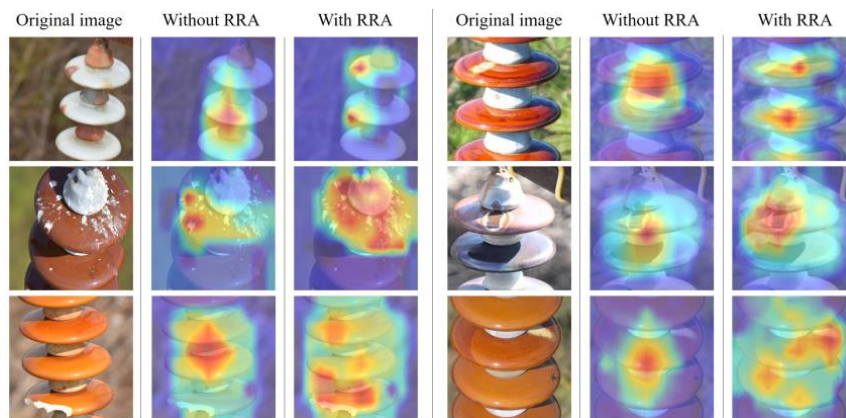


Figure 11. The visual results of Grad-CAM for the RRA.

4.2.3. Sensitivity Analysis of α and β

To better understand the effects of the two balance parameters α and β , in the total loss formula Equation (7), we conducted a sensitivity analysis of α and β . As shown in Figure 12, we set α and β to (0, 1), (0.2, 0.8), (0.4, 0.6), (0.8, 0.2), and (1, 0), respectively. When $\alpha = 0$ and $\beta = 1$, the total loss is optimized only through KL divergence, and the model cannot converge. After increasing α and decreasing β , the accuracy initially increases but then decreases. When $\alpha = 0.8$ and $\beta = 0.2$, the accuracy reaches its peak. We analyze that this is because the semantic information learned by the cross-entropy loss plays a decisive role in the classification performance. When α is too small, the role of cross-entropy loss is weakened, and the accuracy inevitably decreases or even fails to converge. On the other hand, KL divergence plays a supporting role in forcing multi-granularity features to focus on different regions, thereby helping to capture as many details as possible. The experiments show that an appropriate β indeed optimizes the feature extraction of the network.

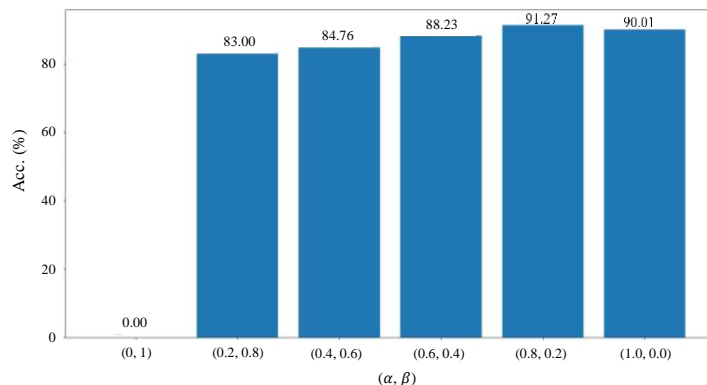


Figure 12. Sensitivity analysis study of α and β .

4.2.4. Sensitivity Analysis of k

To investigate the impact of different values of k in Equation (4) on recognition accuracy, we vary the values of k in a set of {1.0, 1.3, 1.6, 1.9, 2.1}. As shown in the Table 4, the accuracy shows a stable then decreasing trend as k increases. The accuracy reaches its peak when $k = 1.3$. Our analysis shows that reducing the value of k results in less distortion in the insulator patch and improves recognition accuracy. However, this also increases the number of clippings, which slows down recognition speed for a single image. Conversely, increasing k results in more severe distortion, but improves recognition speed for a single image. After considering both accuracy and speed, we determined that $k = 1.3$ is the optimal choice.

Table 4. Sensitivity analysis study of k .

k	Acc (%)						Speed (Images/s)
	Average	Normal	Anomaly	Pollution	Breakage	Thunderstroke	
1.0	91.12	98.28	92.46	87.92	81.99	87.52	95.2
1.3	91.27	98.31	92.50	90.01	82.05	87.99	126.2
1.6	90.82	98.72	92.22	87.69	81.81	86.95	144.1
1.9	89.65	97.31	91.30	86.86	80.25	86.45	156.2
2.1	88.12	95.65	90.28	85.65	79.05	84.60	180.2

4.3. Comparison Experiment

4.3.1. Quantitative Evaluation

According to Tables 5 and 6, our MGFNet achieves an accuracy of 91.27%, 98.31%, 92.50%, 88.00%, 82.05%, and 87.52% on all of class, normal, anomaly, pollution, breakage, and thunderstroke, respectively. It is observed that without the TC, other state-of-the-art methods perform significantly worse in accuracy than the MGFNet. Therefore, we mainly compare these methods with the TC. From the Tables 5 and 6, it can be seen that lightweight methods (such as SqueezeNet, MobileNet, and ShuffleNetv2) have advantages in terms of parameter and speed. However, their accuracy is not sufficient for practical applications. With the development of hardware, the tolerance for model parameters has increased. Our MGFNet outperforms lightweight methods with accuracy and has already met real-time requirements. Recent methods such as CSRA and MobileViTv2 achieve a similar accuracy to the MGFNet on all classes. However, there is still a difference of 5.39–4.14% in accuracy for fine-grained categories (such as anomaly categories), indicating that the proposed MGFNet can better handle these fine-grained features. Although methods such as CSRA and MobileViTv2 improve the feature selection ability by using attention mechanisms, they are still inferior to the MGFNet in the overall accuracy and accuracy of anomaly categories. This may be because these advantage algorithms lack the ability to fuse multi-granularity features. In general, our method is significant in accuracy, especially for anomalous categories. Although our method has not achieved the best performance in terms of the parameter quantity and speed, it has fully met the practical requirements.

Table 5. Comparison of MGFNet state-of-the-art methods on accuracy.

Model	TC	Acc (%)					
		Average	Normal	Anomaly	Pollution	Breakage	Thunderstroke
ResNet50 [20]	No	56.39	70.13	80.49	81.99	30.76	38.89
SqueezeNet [21]	No	46.80	54.16	80.99	79.99	12.82	45.84
MobileNet [22]	No	46.48	54.65	80.25	79.18	12.75	43.97
ShuffleNetv2 [23]	No	46.20	54.15	80.52	79.85	12.75	43.16
SENet [14]	No	59.11	73.24	77.12	83.02	38.02	37.09
CBAM [16]	No	59.75	73.65	78.63	83.45	38.46	38.55
CSRA [24]	No	61.35	76.26	79.36	82.64	39.96	39.91
MobileViTv2 [25]	No	62.32	75.36	79.83	82.36	38.36	48.28
ResNet50 [20]	Yes	88.08	95.12	86.49	89.99	77.23	85.12
SqueezeNet [21]	Yes	72.09	81.94	84.49	73.99	57.69	66.66
MobileNet [22]	Yes	82.26	91.66	85.99	77.99	75.64	73.59
ShuffleNetv2 [23]	Yes	76.16	79.86	82.99	83.99	67.94	72.22
SENet [14]	Yes	89.28	98.45	87.65	88.29	78.12	83.85
CBAM [16]	Yes	89.03	98.22	87.35	88.15	77.99	84.11
CSRA [24]	Yes	89.77	98.56	87.11	88.89	78.25	82.42
MobileViTv2 [25]	Yes	90.07	96.88	88.36	88.23	81.23	87.30
MGFNet (Ours)	Yes	91.27	98.31	92.50	90.01	82.05	87.99

Table 6. Comparison of MGFNet state-of-the-art methods on params and speed.

Model	TC	Params (Megabytes)	Speed (Images/s)
ResNet	Yes	42.5	156.3
SqueezeNet	Yes	1.2	1342.7
MobileNet	Yes	3.5	654.3
ShuffleNetv2	Yes	1.4	503.8
SENet	Yes	44.5	115.1
CBAM	Yes	44.5	125.0
CSRA	Yes	45.7	124.16
MobileViTv2	Yes	19.30	125.3
MGFNet(ours)	Yes	84.1	126.2

4.3.2. Qualitative Evaluation

Figure 13 illustrates the recognition results of various methods on the same targets, providing qualitative results. The samples (a)–(d) in Figure 13 show that the MGFNet, MobileViTv2, CSRA, CBAM, SENet, and ResNet exhibit a superior recognition performance for relatively simple targets. However, SqueezeNet, MobileNet, and ShuffleNetv2 have some target misclassifications, indicating that the robustness of these methods needs improvement. The samples (e)–(h) are more challenging to recognize due to relatively concealed defects, leading to easy network misidentification. For example, only the ResNet and MGFNet correctly identify target (e), but the ResNet has misclassifications in (f)–(h). Target (f) shows that the reflection greatly misleads the algorithm, causing most methods to misjudge it as a damaged insulator. However, the proposed MGFNet model can still correctly identify the target in (f), indicating strong robustness. In sample (h), the similarity between the lightning strike and the normal area makes it easy for even a person to make a wrong judgment, but the MGFNet can still successfully identify it. These results demonstrate that the MGFNet has an excellent feature extraction ability in identifying small defects, emphasizing the effectiveness of the proposed method in improving the recognition performance of defective insulators. However, from the recognition performance of the MGFNet on Figure 12i,j, it can be seen that the MGFNet is not sensitive enough to morphological defects and lacks the ability to extract edge features. Therefore, in future work, we hope to add an edge extraction sub-network based on the existing network to improve its learning ability for morphological features.

4.4. MGFNet-Based Two-Stage Insulator Defect Detection Experiment

Based on Table 7, it can be observed that when the MGFNet is added to the YoLov5 and Faster RCNN, their defect recognition performance is better than using YoLov5 and Faster RCNN alone. This is because recognizing defects using the YoLov5 and Faster RCNN alone requires being faced with complex background interference, while the YoLov5+MGFNet and Faster RCNN+MGFNet can effectively eliminate background interference and achieve more precise recognition of defects. Moreover, based on Table 7, both the YoLov5+MGFNet and Faster RCNN+MGFNet have used the MGFNet for insulator defect detection. However, the YoLov5+MGFNet outperforms the Faster RCNN+MGFNet in terms of defect recognition accuracy with a higher accuracy score of 91.27% compared to 91.17% by the Faster RCNN+MGFNet. In terms of the parameters, the YoLov5+MGFNet has a smaller number of parameters with 98.1 megabytes compared to the Faster RCNN+MGFNet which has 444.1 megabytes. Additionally, the speed of the YoLov5+MGFNet is 24.45 images/sec, while the speed of the Faster RCNN+MGFNet is 16.86 images/sec. Therefore, the YoLov5+MGFNet is a better choice for practical engineering deployment. The detection results of the YoLov5+MGFNet are shown in Figure 14.

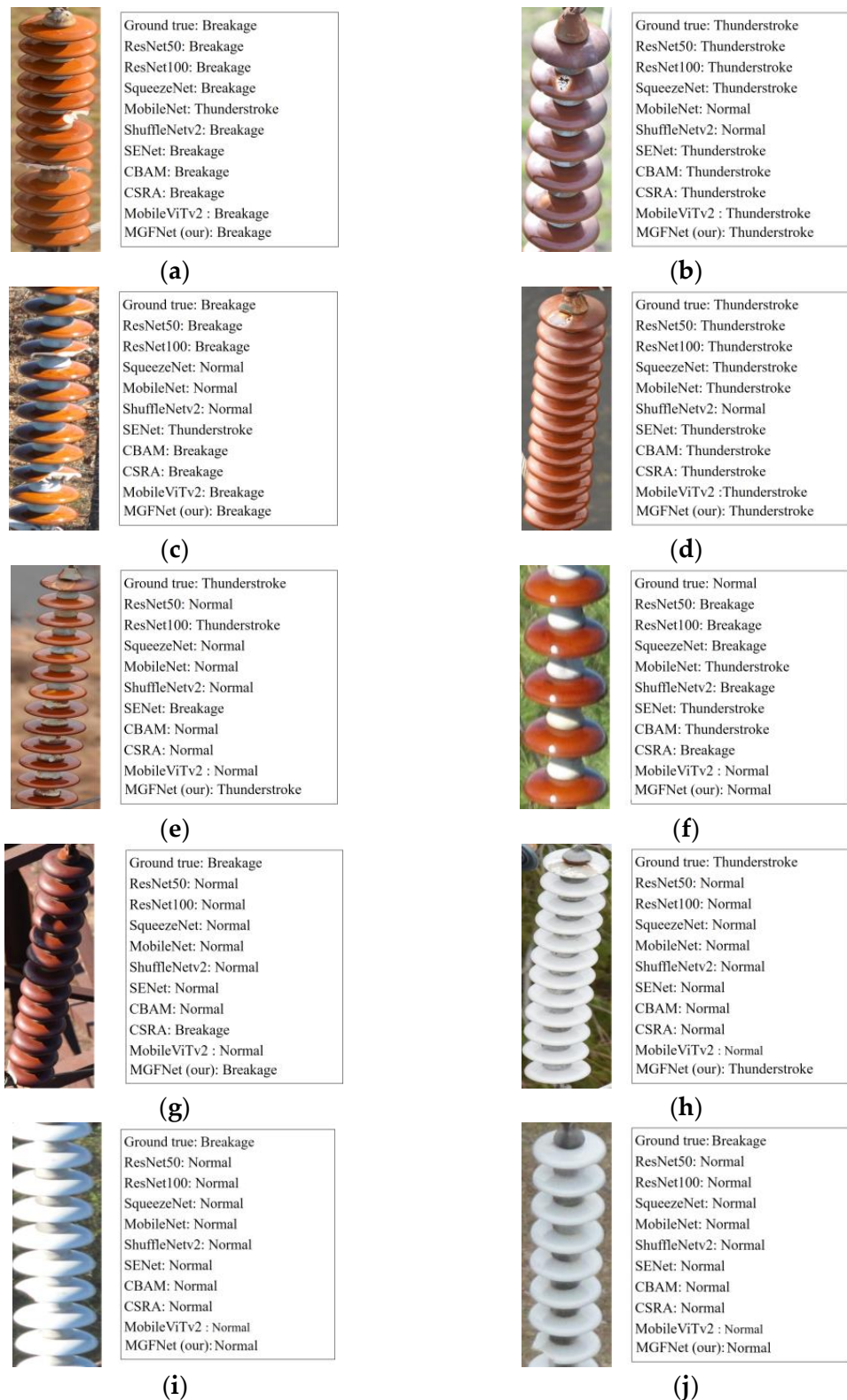


Figure 13. Qualitative comparison. From (a) to (j), the difficulty of recognition increases gradually.

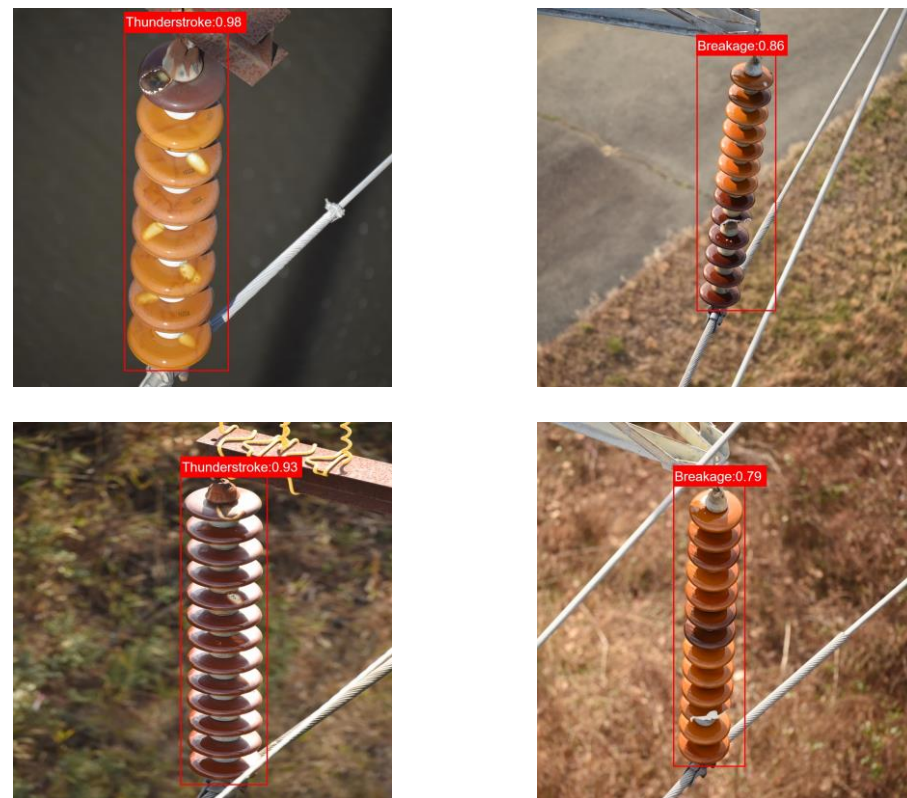


Figure 14. Detection performance of MGFNet-based two-stage insulator defect detection.

Table 7. The performance of MGFNet-based two-stage insulator defect detection.

Model	Insulator Extraction	Defect Recognition	Params (Megabytes)	Speed (Images/s)
	mAP@0.5 (%)	Acc (%)		
YoLov5	100	79.15	14	19.19
Faster RCNN	100	79.85	360.1	29.12
YoLov5+MGFNet	100	91.27	98.1	24.45
Faster RCNN+MGFNet	100	91.17	444.1	16.86

4.5. Performance of MGFNet on UAV Platform

To achieve the real-time detection of an insulator inspection by drones, we utilize the on-board camera ZED-Mini to capture UVA images, and subsequently employ the on-board AI system Nvidia Jetson TX2 (TX2) to run the proposed MGFNet-based two-stage insulator defect detection algorithm to diagnose insulators. The operating system of TX2 is Ubuntu 18.04, using PyTorch version 1.8.0 and Python version 3.6.9. By configuring the deep learning environment on the TX2, we tested the detection speed of the MGFNet running on the drone. Communication is facilitated through the Mavlink drone communication protocol between the communication layer and the ground station or DroneKit. The interface layer functions as the visual interface for the ground station and DroneKit. DroneKit, an open-source software toolkit developed by American 3D Robotics, permits the third-party development of drone applications. As shown in Figure 15, this is the screen of real-time detection by the drone, which successfully determined the health status of the insulator, with a real-time detection speed of 25.25 images/sec. This proves that our proposed MGFNet-based two-stage insulator defect detection algorithm can be effectively deployed on the drone platform.



Figure 15. An example of MGFNet detection on UAV platform.

5. Conclusions

This paper proposes a multi-granularity fusion network (MGFNet) for insulator defect recognition. The MGFNet uses a traversal detection module to alleviate the problem of defect distortion. To better extract the features of subtle defects, a new progressive multi-scale learning strategy is proposed, which integrates the features of different scales and increases the diversity of features. Finally, the RRA module is used to replace and better learn the relationships between non-local features, improving the receptive field of features. Multiple experiments have verified the effectiveness and practicality of our MGFNet. In addition, we have successfully deployed the MGFNet-based two-stage insulator defect detection algorithm on a UAV platform and completed real-time inspection for UAV images.

For future research, we identify two directions. First, some types of insulator defects are extremely rare and cannot be trained with the current networks. Therefore, algorithms capable of classifying with limited samples are needed. Few-shot learning is a new method that utilizes techniques such as meta-learning and generative models to achieve the ability to learn target features from a small number of samples. Second, we can use generative adversarial networks (GANs) to generate rare insulator defect images. A GAN is a generative model that trains a generator and a discriminator to generate data similar to the real data distribution. GAN-based methods use the generator to generate a large amount of data for training the classifier. This method has achieved good results in some tasks and is expected to become a promising method in the future.

Author Contributions: Funding acquisition, Y.L. and F.S.; methodology, Y.L. and Z.L.; software, Z.L.; writing—original draft, Z.L.; writing—review and editing, Y.L. and F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Guangxi Science and Technology base and Talent Project (Grant No. Guike AD22080043) the Natural Science Foundation of Guangxi under Grant 2022GXNSFBA035661, and the Hubei Key Laboratory of Intelligent Robot (Grant No. HBIR202108). The APC was funded by Feng Shuang and Yong Li.

Data Availability Statement: The data in this paper are undisclosed due to the confidentiality requirements of the data supplier.

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

References

1. Liu, W.; Liu, Z.; Wang, H.; Han, Z. An automated defect detection approach for catenary rod-insulator textured surfaces using unsupervised learning. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 8411–8423. [[CrossRef](#)]
2. Zhang, D.; Gao, S.; Yu, L.; Kang, G.; Wei, X.; Zhan, D. DefGAN: Defect detection GANs with latent space pitting for high-speed railway insulator. *IEEE Trans. Instrum. Meas.* **2020**, *70*, 1–10. [[CrossRef](#)]
3. Song, Y.; Liu, Z.; Ling, S.; Tang, R.; Duan, G.; Tan, J. Coarse-to-Fine Few-Shot Defect Recognition with Dynamic Weighting and Joint Metric. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 2514910. [[CrossRef](#)]
4. Liu, X.; Miao, X.; Jiang, H.; Chen, J. Box-point detector: A diagnosis method for insulator faults in power lines using aerial images and convolutional neural networks. *IEEE Trans. Power Deliv.* **2021**, *36*, 3765–3773. [[CrossRef](#)]
5. Zhang, X.; Zhang, Y.; Liu, J.; Zhang, C.; Xue, X.; Zhang, H.; Zhang, W. InsuDet: A fault detection method for insulators of overhead transmission lines using convolutional neural networks. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 5018512. [[CrossRef](#)]
6. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
7. Lan, Z.; Hong, Y.; Li, Y. An improved YOLOv3 method for PCB surface defect detection. In Proceedings of the 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA), Shenyang, China, 22–24 January 2021; pp. 1009–1015.
8. Zhao, Y.J.; Yan, Y.H.; Song, K.C. Vision-based automatic detection of steel surface defects in the cold rolling process: Considering the influence of industrial liquids and surface textures. *Int. J. Adv. Manuf. Technol.* **2017**, *90*, 1665–1678. [[CrossRef](#)]
9. Sharifzadeh, M.; Alirezaee, S.; Amirfattahi, R.; Sadri, S. Detection of steel defect using the image processing algorithms. In Proceedings of the International Conference on Electrical Engineering, Cairo, Egypt, 27–29 May 2008; pp. 125–127.
10. Leavers, V.F. Which hough transform? *CVGIP Image Underst.* **1993**, *58*, 250–264.
11. Yang, J.; Li, X.; Xu, J.; Cao, Y.; Zhang, Y.; Wang, L.; Jiang, S. Development of an optical defect inspection algorithm based on an active contour model for large steel roller surfaces. *Appl. Opt.* **2018**, *57*, 2490. [[CrossRef](#)] [[PubMed](#)]
12. Liao, G.-P.; Yang, G.-J.; Tong, W.-T.; Gao, W.; Lv, F.-L.; Gao, D. Study on Power Line Insulator Defect Detection via Improved Faster Region-Based Convolutional Neural Network. In Proceedings of the 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT), Dalian, China, 19–20 October 2019; pp. 262–266. [[CrossRef](#)]
13. Perera, P.; Nallapati, R.; Xiang, B. One-class Novelty Detection Using GANs with Constrained Latent Representations. *arXiv* **2019**, arXiv:1903.08550.
14. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 23 June 2018; pp. 7132–7141. [[CrossRef](#)]
15. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Network. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 19 June 2020; pp. 11531–11539. [[CrossRef](#)]
16. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
17. Li, X.; Wang, W.; Hu, X.; Yang, J. Selective Kernel Networks. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 510–519. [[CrossRef](#)]
18. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Part I 14. Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
19. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via GradientBased Localization. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 618–626.
20. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
21. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
22. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
23. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
24. Zhu, K.; Wu, J. Residual Attention: A Simple but Effective Method for Multi-Label Recognition. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 184–193.
25. Sachin, M.; Rastegari, M. Separable self-attention for mobile vision transformers. *arXiv* **2022**, arXiv:2206.02680.

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.