TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 14/20 pp1796–1809
DOI: 10.26599/TST.2023.9010137
Volume 29, Number 6, December 2024

Enhancing Power Line Insulator Health Monitoring with a Hybrid Generative Adversarial Network and YOLO3 Solution

Ramakrishna Akella, Sravan Kumar Gunturi*, and Dipu Sarkar

Abstract: In the critical field of electrical grid maintenance, ensuring the integrity of power line insulators is a primary concern. This study introduces an innovative approach for monitoring the condition of insulators using aerial surveillance via drone-mounted cameras. The proposed method is a composite deep learning framework that integrates the "You Only Look Once" version 3 (YOLO3) model with deep convolutional generative adversarial networks (DCGAN) and super-resolution generative adversarial networks (SRGAN). The YOLO3 model excels in rapidly and accurately detecting insulators, a vital step in assessing their health. Its effectiveness in distinguishing insulators against complex backgrounds enables prompt detection of defects, essential for proactive maintenance. This rapid detection is enhanced by DCGAN's precise classification and SRGAN's image quality improvement, addressing challenges posed by low-resolution drone imagery. The framework's performance was evaluated using metrics such as sensitivity, specificity, accuracy, localization accuracy, damage sensitivity, and false alarm rate. Results show that the SRGAN+DCGAN+YOLO3 model significantly outperforms existing methods, with a sensitivity of 98%, specificity of 94%, an overall accuracy of 95.6%, localization accuracy of 90%, damage sensitivity of 92%, and a reduced false alarm rate of 8%. This advanced hybrid approach not only improves the detection and classification of insulator conditions but also contributes substantially to the maintenance and health of power line insulators, thus ensuring the reliability of the electrical power grid.

Key words: DCGAN; generative adversarial networks; insulators; SRGAN; YOLO

1 Introduction

Regular inspection of electrical utility equipment is essential for maintaining system reliability and minimizing disruptions. Traditional methods, such as manual patrolling, are time-consuming and labor-intensive. Insulators, crucial for grid protection, require efficient and cost-effective inspection techniques[1]. These insulators, exposed to diverse environmental conditions, face challenges such as decreased surface resistance and increased susceptibility to over-voltage and puncture due to ongoing leakage currents[2, 3]. High humidity and exposure to corrosive environments further exacerbate their vulnerability, leading to potential flash-overs[4, 5]. The failure of high-voltage insulators can lead to significant network outages and considerable financial losses.

1.1 Advancements in insulator health monitoring

The necessity for efficient insulator inspection has led

• Ramakrishna Akella and Sravan Kumar Gunturi are with Department of Electronics and Communication Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad 500075, India. E-mail: ramakrishna.a@kluniversity.in; sravankumar.gunturi@gmail.com.
• Dipu Sarkar is with Department of Electrical and Electronics Engineering, National Institute of Technology Nagaland, Dimapur 797103, India. E-mail: dipusarkar5@rediffmail.com.
* To whom correspondence should be addressed.
Manuscript received: 2023-08-20; revised: 2023-12-14; accepted: 2023-12-26

to the development of innovative monitoring techniques. Traditional manual inspections, though essential, are impractical for extensive and challenging landscapes. Thus, image classification-based video surveillance has emerged as a more effective monitoring approach. Recent research has focused on automating the assessment of high-voltage insulators. The early "Buzz method"[6], which involves physical inspection under high voltage, has given way to more advanced techniques. These include the analysis of partial discharge-generated electromagnetic waves using high-frequency signals (30–300 MHz)[7]. The fractal algorithm is now employed to analyze frequency spectra from fast fourier transform for detecting insulator defects. Moreover, concerns such as the impact of algae fungus on outdoor electrical grid insulation have been identified[8]. Recent proposals include a hybrid method combining wavelet analysis with support vector machines for monitoring insulator health[9], and similar techniques have been applied for detecting insulation damages in large generators[10]. The introduction of classifiers like naive Bayes and more advanced systems such as support vector machines (SVM) and random-forest has further enhanced the efficiency of insulation damage detection[11, 12].

1.2 Deep learning and UAVs in insulator monitoring

To overcome the limitations of traditional machine learning, deep learning algorithms have been introduced for object extraction and classification from images[13]. Aerial surveillance using unmanned aerial vehicles (UAVs) has emerged as an effective method for insulator health monitoring, offering advantages in terms of flexibility, low operation costs, and advanced computer vision for real-time analysis[14, 15]. Despite their effectiveness, deep learning methods require extensive datasets, necessitating various data augmentation strategies, such as image modification[16]. Singh et al.[17] presented a computer vision system that utilized infrared thermal (IRT) cameras to automate the examination of power line insulators, thereby minimizing the requirement for dangerous hand inspections. The system utilizes machine learning, specifically a Gaussian kernel support vector machine, to accurately categorize defects in insulators based on IRT images. The study conducted by Stefenon et al.[18] presented a novel hybrid approach for forecasting failures in electric power distribution insulators by analyzing leakage current time series. Utilizing the Christiano–Fitzgerald random walk filter and the group data-handling mechanism, this method surpasses conventional models in terms of accuracy. However, a primary limitation of the research is its controlled laboratory environment, which may not fully capture the intricacies of the real world. Despite this, the approach shows considerable potential in improving the reliability of power supply by proactively identifying faults.

1.3 Limitations of current methods

Most of the methods mentioned above have the following limitations:

- Manual inspection involves much expense and risk.
- The size of the dataset affects the accuracy of deep learning algorithms.
- Low-resolution (LR) drone images reduce the detection efficiency of deep-learning-assisted techniques.
- The accuracy of detection and speed of current deep learning-based object-detection algorithms are poor.

The major contributions of the current study are fourfold: (1) The resolution of low-quality aerial images is improved by using super-resolution generative adversarial networks to increase the accuracy. (2) The data size problem is solved by creating fake images using deep convolutional generative adversarial networks (DCGAN). (3) Advanced object detection models to recognize damages in the insulator are evaluated and compared. (4) The highest detection rate is obtained using the suggested hybrid model. The remainder of the article is organized as follows: Section 2 explains about theoretical background of advanced deep learning models. The implementation of the suggested model is discussed in Section 3. Section 4 details the experiment results and analysis. Section 5 summarizes the paper with essential directions for future work.

2 Theoretical Background

Developments in computer vision have improved classification and object detection efficiencies significantly, especially in convolutional neural networks[19, 20]. Graphical processing systems have also contributed greatly to convolutional neural networks

(CNN) applications by using parallel operations to address data-intensive operations issues in real-time. The proposed hybrid detector consists of two distinct generative adversarial network models together with an advanced object detection model called "You Only Look Once" version 3 (YOLO3).

2.1 Generative adversarial network

The new advancements in data-augmentation enables the generation of synthetic data from high-quality images. Generative adversarial network (GAN) is a promising game theory-inspired method of generating synthetic images[21]. Since GANs gained significant prominence in computer vision, researchers started using different GANs to produce high-quality images. Latest GAN's applications include image-to-image conversion[22], text-to-image conversion[23], and 3D point cloud inpainting[24]. In this experiment, we used one of the GAN variants to produce synthetic insulator images to solve the imbalanced learning problem. Due to the instability and intractability of the original GAN model, over the period, DCGAN is frequently using with good results. As shown in Fig. 1, DCGAN has two separate networks, G is a generator network, receiving random noise Z as input and generating images through this noise. D is a network of discriminators that decides whether an image is real or not. The parameter of its information is X, which represents an image. D production represents the probability of actual photos. In short, the discriminator is a binary classifier that produces 0 for the fake image and 1 for the exact image. Here, G and D are typically non-linear functions of mapping, such as convolutionary neural networks.

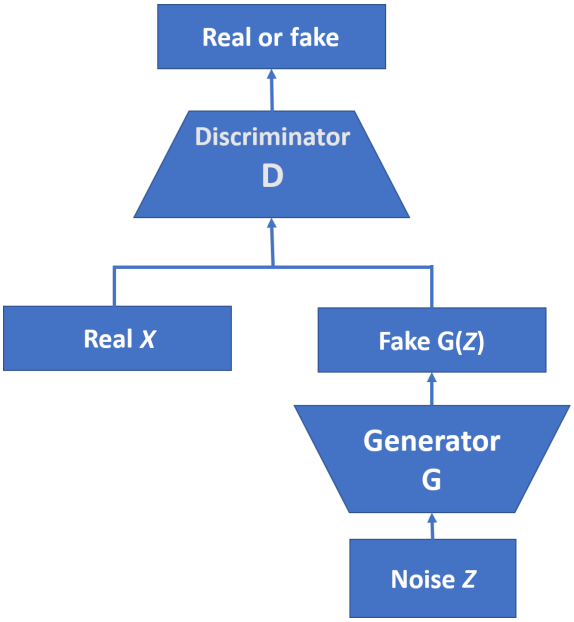

Fig. 1 DCGAN architecture.

2.2 Image super-resolution

UAV image capabilities offer many benefits, but few critical concerns regarding image resolution must be acknowledged. First, factors such as the camera quality, exposure times, movement, and environment's impact degrade aerial image resolution. Second, flight altitude is often increased to reduce the time required to take wide-field pictures, resulting in low-resolution images. Extracting information from low-resolution images is not reliable, and most decisions must be made based on blurred images. Therefore, the authors proposed two models to increase the resolution of low-quality aerial photos, one with CNN (SRCNN) and the other with GAN (SRGAN). The purpose for SRCNN and SRGAN models is to retrieve finer textures from the picture as we scale it up so that its clarity can not be affected. There are other techniques, such as Bi-linear interpolation[25], that can be used to perform this job, but they suffer from loss of image information and smoothing. From the proposed system, the authors found that the SRGAN has the highest peak signal-to-noise ratio and produces a more eye-friendly picture compared to SRCNN.

2.3 Object detection model

Object detection means to identify objects in the image and to classify them by type. In this work, we have chosen three modern object detectors: faster-RCNN, YOLO2, and YOLO3. Redmon et al.[26] suggested a novel object detection model known as YOLO: You Only Look Once. It transfers the $n\times n$ image only once in a fully convolutional neural network, which makes it quite fast. Later, it splits the entire image into grids of size $m\times m$ and produces bounding boxes and their class probabilities. However, localization errors are high because of grid processing, and the accuracy is also low. To resolve the above issues, by adopting a batch-normalization to the convolution, YOLO2 is suggested to improve detection efficiency[27]. YOLO2 also provides an anchor-box, various levels of preparation, and high-quality features. The detection accuracy for small items, however, remains low. Redmon and Farhadi[28] have thus introduced YOLO3, which has a deep network of convolution layers for better precision.

To resolve the vanishing gradient problem, it utilizes a residual hop relationship. YOLO3 estimates boxes on three separate scales then obtain attributes from those scales as used in the pyramid network function[29]. The network's prediction result is a bounding box, object score, and class prediction. This process helps YOLO3 to monitor objects of various sizes. Therefore, YOLO3 is appropriate for precision and speed object recognition applications.

3 Proposed Work

This section discusses the working of a suggested hybrid model for state evaluation of the insulators with the aid of the block diagram, as shown in Fig. 2. Firstly, the source image data-set is applied to an image processing unit where blurred images are separated from clear photos. We used the Laplace distribution variance to distinguish the blurry low-resolution from clear high-resolution (HR) images. To improve the contrast between the adjacent image details, the Laplace operator selects a second-degree image differential. Essentially, the operator is initially used to modify the picture and then to ascertain the variance. The boundary is more appropriate in clear pictures, so that the difference increases considerably. In contrast, edge information is comparatively smaller in blurred images, so is the distinction. Whereas if the variance is much smaller than the stated threshold, the image is classified as blurry. Otherwise, the provided picture is referred to as clear. Thus the surveillance photos are subdivided into blurred and clear groups of images. These blurry pictures are employed as inputs to the SRGAN, transforming them into images with super-resolution. Finally, the converted images are added to the original image data collection used as a new database for further analysis. Next, a new image data-set is used as DCGAN training input to create fake images. These photos are merged with the previous clearer images to create a complete dataset of the image. The manual annotation process begins after obtaining the final image dataset and locates the desired insulator objects. The annotated images are given as training data for the YOLO3 object detection model. The input images have now been resized to 416 × 416 to quicken the learning job. The Darknet-53 extract attributes of insulator damages when fed with resized images. The feature-pyramid-network (FPN) approach produces predictions across three separate stages. Mostly, the bounding box parameters, object performance measure, and classification type are found in YOLO3 forecasts. As shown in Fig. 3, the method of removing low confidence boxes is considered to be a

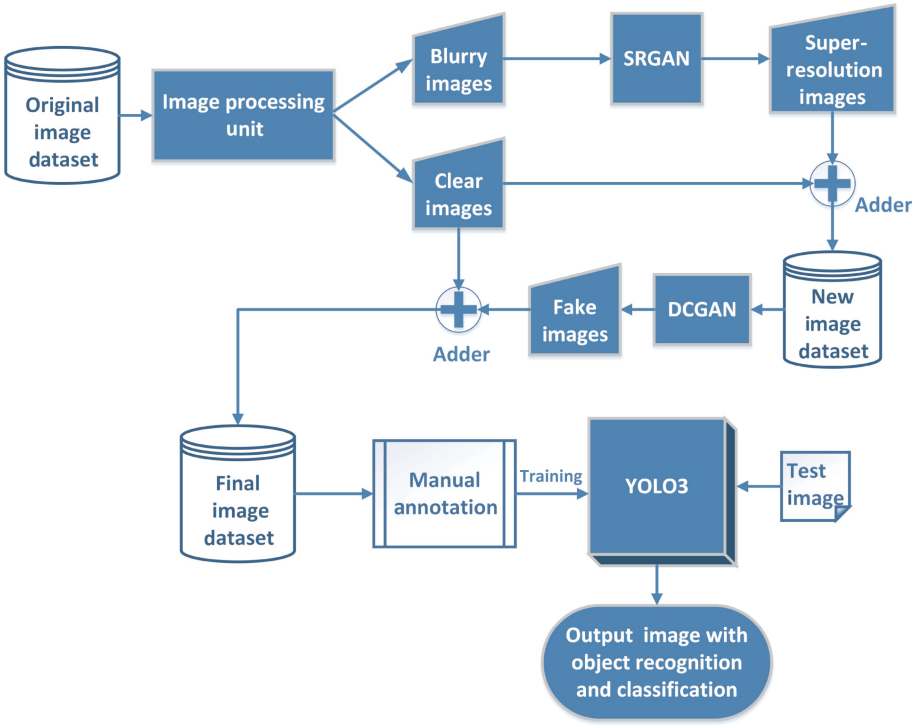

Fig. 2 Block schematic of the proposed model.

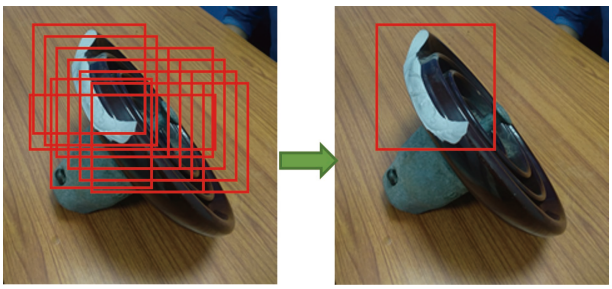

Fig. 3 Non-maximum suppression process.

non-maximum suppression process. Next, under several thresholds, YOLO3 selects the connectors that cover the subject of ground-truth, and then gives each bounding box and associated positioning classification.

3.1 Deep learning models for super-resolution

The advanced deep learning models such as SRCNN and SRGAN are chosen to transform low-resolution images into high-resolution ones. The following subsections provide a detailed description of the above-said models.

3.1.1 SRCNN

Dong et al.[30] have introduced a novel deep learning model called SRCNN, which addresses low-resolution image problems. Usually, the above model consists of three processes: (1) Extraction of patches: This method selects patches from the blurred image P (low-resolution) and describes each patch as higher-order vector space. These vectors are composed of a set of characteristic maps that are equivalent to the vector dimension. The convolution product contains the attributes, and the rectified linear unit obtains the final picture of the primary convolution level. (2) Non-linear mapping: This procedure maps every higher-ordered vector onto another non-linear higher-ordered vector. Intuitively, each mapped vector is another set of feature maps comprised of these vectors. (3) Rebuilding: Here, the convolution layer integrates all super-resolution frames generated to create a super-resolution model, the resultant image performance of the SRCNN network. As a loss function, the mean squared error (MSE) considered from Ref. [30] is used.

3.1.2 SRGAN

The SRGAN architecture includes two parts generator and discriminator, identical to GAN architecture, where the generator produces certain results dependent on the distribution of probability and discriminator attempts to guess weather data from input dataset or generator. The generator and discriminator architecture used in this analysis are derived from Ref. [31]. Generator always attempts to refine the data produced such that the discriminator can be fooled. The generator contains a residual network instead of a deep convolution network. The residual networks are simple to train and enable them to be considerably deeper to achieve better performance. That is because the residual network used a form of connection called skip connections. The residual blocks, which derive from ResNet, are presented in architecture. Two convolutionary layers are used inside the residual block, supported by normalization structures and ReLU as an activation, with a tiny 3×3 kernels and 64 characteristic vectors. The resolution of the source images is enhanced with two guided sub-pixel convolution layers. Instead of using a fixed value for an alpha parameter like LeakyReLU, this generator architecture also uses parametric-ReLU (activation function). It learns the parameters of the rectifier adaptively and increases the precision at a negligible extra computational expense. A high-resolution image is under-sampled to a low-resolution picture throughout the training.

The architecture of the generator then aims to up-sample the picture from low-resolution to super-resolution. The picture is then transferred into the discriminator, the discriminator, which attempts to differentiate between a super-resolution and an HR image and produce the adversarial loss that then back-propagated into the architecture of the generator. Discrimination between actual HR images and produced SR images is the role of the discriminator. The architecture of the discriminator used in this paper is identical to the architecture of DCGAN with LeakyReLU as activation. Eight convolutional layers of 3×3 filtering kernels are used in the network, rising from 64 to 512 kernels by a factor of 2. Strided convolutions are used every time the characteristics count is doubled to decrease the image resolution. Two thick layers and a LeakyReLU added between the resulting 512 attribute maps and a last sigmoid activation unit are followed to achieve a likelihood for classification. The perpetual loss function is used in SRGAN, which is the weighted total of the adversarial and content losses since it is needed to ascertain the generator's performance. We followed the concept of perceptual loss from C. Ledig et al.[31] In the current

work, the pixel wise MSE loss is measured by using the following equation:

$$l_{\mathrm{MSE}}^{\mathrm{SR}} = \frac{1}{r^2 WH} \sum_{j=1}^{rW} \sum_{k=1}^{rW} (I_{j,k}^{\mathrm{HR}} - G_{\theta G}(I^{\mathrm{LR}})_{j,k})^2 \qquad (1)$$

where $l_{\mathrm{MSE}}^{\mathrm{SR}}$ lSR MSE is the pixel-wise mean squared error loss for super-resolution, quantifying the discrepancy between the high-resolution images produced by the generator and the actual high-resolution images. The upscaling factor, r, determines the enlargement of the low-resolution input images (I^{LR}) dimensions to generate super-resolution images. W and H denote the width and height, respectively, of the highresolution images (I^{HR}), used for normalizing the loss calculation. The indices j and k iterate over the width and height of the upsampled image, ensuring the loss is computed across each pixel by comparing the pixel value at position (j, k) in the actual high-resolution image, $I_{j,k}^{\mathrm{HR}}$, against the corresponding pixel value in the super-resolution image generated by the generator (G_θ), $G_{\theta G}(I^{\mathrm{LR}})_{j,k}$. The squared term ensures the error is positive and emphasizes larger discrepancies between the generated and actual high-resolution images.

3.2 DCGAN

GAN, an advanced neural network, often tries to learn about the distribution of raw data. The system generator (G) establishes a mapping correlation between the Gaussian distribution and the actual distribution of results following the direction of a discriminatory network (D). Deep convolutionary generative adversarial networks is one of GAN's most popular and useful network maps, consisting of convolution stages with no fully-connected or max-pooling layers. It employs convolution layers and conversion for down-sampling and up-sampling to facilitate network training from model structure optimization.

In GAN networks' training, two contrasting D and G networks have specific weights and bias sets. By improving its parameters, the discriminator's optimization mechanism minimizes its loss function $L_{\mathrm{D}}(\theta,\phi)$. Likewise, G associated optimization method is constructed by changing its parameter sets to reduce its loss function $L_{\mathrm{G}}(\theta,\phi)$. The generator's purpose is to learn how to distribute original data and then produce samples as close as possible to the actual representation. The generator input is random noise $Z = (Z^1, Z^2, ..., Z^m)$, and network output is integrated sample order $\boldsymbol{G}(Z) = (\boldsymbol{G}(z)^1, \boldsymbol{G}(z)^2, ..., \boldsymbol{G}(z)^m)$. The discriminator D goal is to detect whether the input data are the original data X or $\boldsymbol{G}$'s output series $\boldsymbol{G}$(Z). If the input is the actual X, D gives one as output. Otherwise, it provides 0 output. GAN's loss is as expressed in Eq. (2), and the purpose of its optimization is given in Formula (3).

$$L(D,G) = E_{x\sim P_{\mathrm{data}}(x)}[\log D(x)] + E_{z\sim P_z(z)}[\log(1 - D(\boldsymbol{G}(z)))] \qquad (2)$$

$$\min_G \max_D L(D,\boldsymbol{G}) \qquad (3)$$

Equation (2) defines the loss function $L(D,G)$ of a generative adversarial network, consisting of two terms. The first term, $E_{x\sim P_{\mathrm{data}}(x)}[\log D(x)]$, represents the expected log-likelihood for the discriminator D to correctly identify real data samples x drawn from the data distribution $P_{\mathrm{data}}(x)$. The second term, $E_{z\sim P_z(z)}[\log(1 - D(\boldsymbol{G}(z)))]$, represents the expected log-likelihood for the discriminator to incorrectly classify fake data samples produced by the generator G. These fake samples are generated from noise variables z drawn from the noise distribution $P_z(z)$. The discriminator's goal is to maximize this loss function by assigning the correct labels to both real and fake samples, while the generator's goal is to minimize the second term of the loss by generating samples that are indistinguishable from real data.

Formula (3) summarizes the adversarial training objective, where the generator G aims to minimize the loss function $L(D, G)$ against an adversary, the discriminator D, that seeks to maximize it. This minmax game leads to the training of both the generator to produce data resembling the real data distribution, and the discriminator to distinguish between real and generated data effectively.

For the training of Generator networks, the objective is to maximize the probability of the discriminator being fooled into believing that the generated data is real. This goal translates into minimizing the following objective, as depicted in Formula (4):

$$\min_{\boldsymbol{G}} E_{z\sim P_z(z)}[\log(1 - D(\boldsymbol{G}(z)))] \qquad (4)$$

This Formula represents the generator's effort to produce data that the discriminator will classify as real, or in other words, to make $D(\boldsymbol{G}(z))$ as close to 1 as possible.

When training the discriminator D, the loss function is divided into two parts: one that deals with real data

samples and another that pertains to the data generated by the generator $\boldsymbol{G}$. For real data samples X, the discriminator's output $D(X)$ is optimized to be close to 1, indicating that it recognizes the data as real. Conversely, for the fake data generated by $\boldsymbol{G}$, the discriminator aims to output values close to 0, indicating its recognition of the data as fake. The optimization function capturing both these objectives is presented in Formula (5):

$$\mathrm{Max}_D\ E_{x\sim P_{\mathrm{data}}(x)}[\log D(x)] + E_{z\sim P_z(z)}[\log(1 - D(\mathbf{G}(z)))] \tag{5}$$

This equation encapsulates the discriminator's dual objective: correctly identifying real data as real (maximizing $E_{x\sim P_{\mathrm{data}}(x)}[\log D(x)]$) and generated data as fake (maximizing $E_{z\sim P_z(z)}[\log(1 - D(\boldsymbol{G}(z)))]$), as outlined in Goodfellow et al.[19].

3.3 YOLO3: Object detection model

Object detection is detecting objects in a digital image of a particular class inside a frame. A recent deep learning technique, Selective-search, has been used in models such as R-CNN and fast R-CNN to reduce the bounding box count. Multi-stage scanning of images using window-like sliding methods is another technique called over feat. On the other hand, YOLO3 tackles the problem of target identification differently. This design uses an input image only once during the entire process. A single-shot detector is another algorithm for object recognition, but it is much slower than YOLO3 and also failed to detect smaller objects. The YOLO family is a set of in-depth learning models designed to recognize objects quickly. A functional neural net, formerly known as a GoogleNet, later updated as a Darknet, splits an input image into a cell grid. The bounding box and object classification for all type of objects are precisely predicted as shown in Fig. 4. To estimate boxes on different scales, YOLO3 uses the FPN process. It employs a specific collection of convolution plus extra layers known as residual-layers to perform the prediction process and handles the complete image's characteristics to find each least bounding-box. FPN calculates the implementation of full training for every class of bounding boxes, those with average precision and outstanding performance. The YOLO3 process begins by dividing the input inspection picture into $N \times N$ segments and inserting a bounding box anchor for any ground-based truth on the map. Each bounding box specifies four attributes (T_x, T_y, T_w, T_h) for the model network. The method of bounding box generation is considered from Ref [32]. A method is then implemented to predict four co-ordinates: the two middle co-ordinate points of the bounding box (b_x, b_y), the grid (C_x, C_y), the bounding box height b_{h}, and the width b_{w}.

The calculation of the Intersection-over-Union (IoU) metric, which is a standard measure for evaluating the accuracy of object detection models, is based on comparing the areas of the smallest bounding rectangles for detected objects and their ground-truth counterparts. This comparison is concisely expressed in

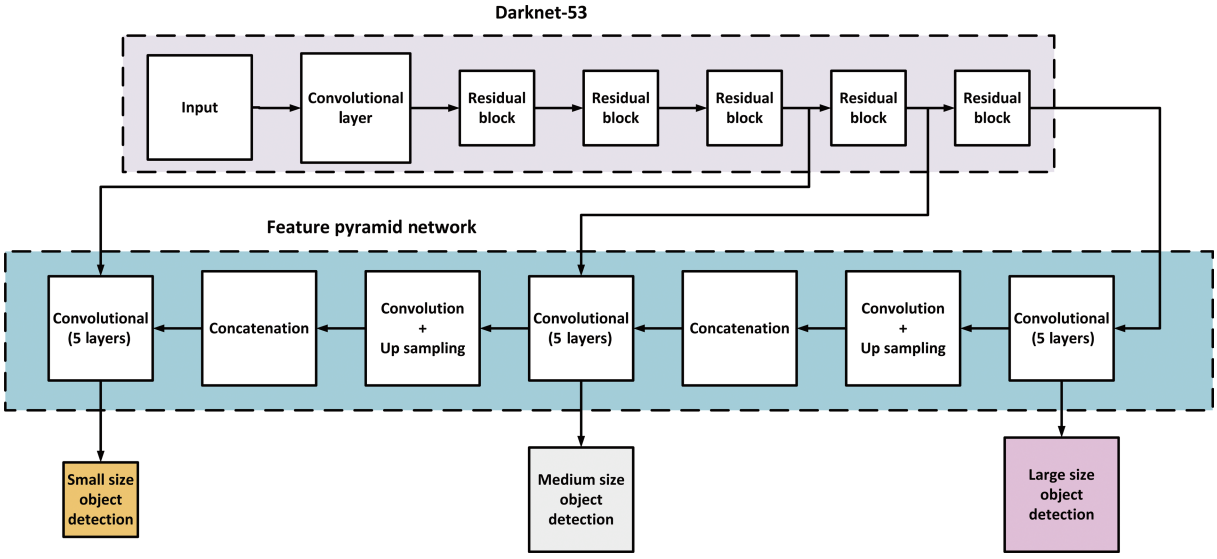

Fig. 4 YOLO3 configuration.

Eq. (6), sourced from Ref. [32]:

$$\text{IOU} = \frac{\text{area}(\text{BB}_{\text{dt}} \cap \text{BB}_{\text{gt}})}{\text{area}(\text{BB}_{\text{dt}} \cup \text{BB}_{\text{gt}})} \tag{6}$$

In this equation, IOU quantifies the overlap between the detected bounding box (BB_{dt}) and the ground-truth bounding box (BB_{gt}). The numerator represents the area of overlap between the two bounding boxes, while the denominator accounts for the total area covered by both bounding boxes, excluding the overlap. Thus, the IoU metric effectively measures the difference between the bounding box detected by the object detection model and the actual ground-truth bounding box, with values closer to 1 indicating higher accuracy in detection.

Network structure of YOLO3. The critical YOLO3 configuration adopts the Darknet-53 structure, as shown in Fig. 4. This model is a fusion of YOLO2, ResNet, and Darknet-19. So, YOLO3 essentially uses convolution kernels of 1×1 and 3×3 and several associated shortcut compositions. Initially, the input inspection picture is processed, then its size is reduced to 416×416 and then converted using YOLO3.

(1) Two layers of convolution type are composed of the first segment. The source image's dimension is 3×416×416, and the kernel sizes are 3×3×32 and 3×3×64. After completion of the convolution phase, the output match type is decreased to 64×208×208.

(2) Three layers of convolution consist of the second section, supported by a residual stage. The kernel size is 1×1×32, 3×3×64, and 3×3×128, and with the subsequent completion of the convolution phase, the output picture is limited to 104×104×128.

(3) Next, the third division comprises of five convolution layers, including two residual type layers. The kernel scale is 1×1×64, 3×3×128, and 3×3×256, and after the convolution phase, the output function estimate is decreased to 52×52×256.

(4) 17 convolution layers with eight residual layers are composed of the fourth segment. The kernel size is 1×1×128, 3×3×256, and 3×3×512, and after completion of the convolution task, the output map is decreased to 26×26×512.

(5) Eight residual layers, along with 17 convolution layers, are used in the fifth layer. The size of the kernel is 1×1×256, 3×3×512, and 3×3×1024. After the convolution procedure has been completed, the output feature map is diminished to 13×13×1024.

(6) Eight convolution layers and four residual layers are composed of the sixth segment. The convolution's sizes are 1×1×512 and 3×3×1024, the convolution stage is completed, and the output map remains the same.

(7) The last segment is made up of three prediction networks. At three different points, YOLO3 forecasts rectangular boxes and then extracts the features of these scales. A 10×10×(3×(4+1+2), tensor for four least bounding box corrections, one object projection, and two classifiers are estimated.

Training. YOLO3 network training is divided into three stages. In phase 1, the surveillance image (size: 5280×2970) taken by the monitoring device is extremely broad to be the system's input. Thus, the image dimension is reduced to 416×416, to accelerate the process of preparation. In phase 2, VOC2007's dataset pattern is used to label good and bad insulators. Finally, Step 3 initializes the network parameters of the YOLO3 model and trains the network to acquire variables to recognize specified objects.

Essential parameters. This article presents a further investigation of the selection of three principal parameters. Batch size: Whenever the more extensive the batch size, the faster it is to prepare. However, we cannot raise the value perpetually because of hardware constraints, so the authors tried out four separate sizes 8, 16, 64, and 128, respectively. When 64, 16, and 8 batch sizes were selected during the preparation process, we wouldn't lose any power; consequently, we picked 64 as the batch size based on the reasons mentioned above. Weight decay: We fixed the correct learning pace and then adjusted the decay model from a constant value (0.01) to the final estimate (0.0005) so that over-fitting can be avoided. Ignore-thresh: The IOU threshold value specifies the sum of IOUs applied in the loss calculation. If the predefined threshold is lower, it results in under-fit and causes over-fitting if the predefined threshold is high. The threshold value for disregard is then set to 0.65 based on the above statement and the case concerned. For classification loss, YOLO3 uses a discrete cross-entropy loss for each set during analysis, eliminating the MSE widely used in previous versions. The loss factor being used conduct the YOLO3 workout is as follows:

$$\text{Loss}(s_m) = \begin{cases} -\log_2(s_m), & \text{if } g_m = 1; \\ -\log_2(1 - s_m), & \text{if } g_m = 0 \end{cases} \tag{7}$$

When m shows the sample's count, $s_m \in [0,1]$ represents the value of the object predicted by the

system, which calculates the expected likelihood that the sample m-th will damage the WT turbine blade. Besides, the truth of the land is indicated by g_m. It should be noted that when the observation of mth applies to the object class, $g_m \in (0,1)$ is inferred. Network variables are trained, i.e., $\sum m\text{Loss}(s_m)$, by minimizing losses for all samples.

4 Result and Discussion

4.1 Hardware configuration

The personal computer's general requirements in this analysis are Intel(R) Xeon(R) CPU E5-2780, NVIDIA 1080 RTX as GPU, primary frequency 2.80 GHz. Operating software is an open source Ubuntu 16.04, and Tensorflow is the algorithm development platform.

4.2 Dataset

No public datasets for insulator groups are available. As a consequence, the authors have experimented with own private dataset. The proposed model is based on a private dataset containing 1000 drone-captured insulator photographs. For which 700 photos have a good insulator. The remaining 300 images contain a bad insulator as an object. From the dataset, 75% of the photographs are employed for training purposes, and 25% are used for validation purposes. The detailed class distribution of the pictures is shown in Table 1.

4.3 Methodology

In this section, a detailed overview of the proposed hybrid model for insulator damage identification is provided. The approach combines SRGAN, DCGAN, and YOLO3 for object detection. The hyperparameters, training procedures, and solutions to challenges encountered during implementation are presented. Figure 5 illustrates the comprehensive data flow steps implemented in the innovative insulator monitoring system. Each step is strategically designed to enhance the efficiency and accuracy of monitoring power line insulators, ensuring the reliability of the electrical power grid. The process encompasses real-world data acquisition through aerial surveillance, advanced pre-processing techniques for image refinement, and the utilization of cutting-edge deep learning models.

Table 1 Insulator class distribution.

Model	Class-good	Class-bad	Total
YOLO3	500	300	800
SRGAN+YOLO3	600	400	1000
SRGAN+DCGAN+YOLO3	1600	1400	3000

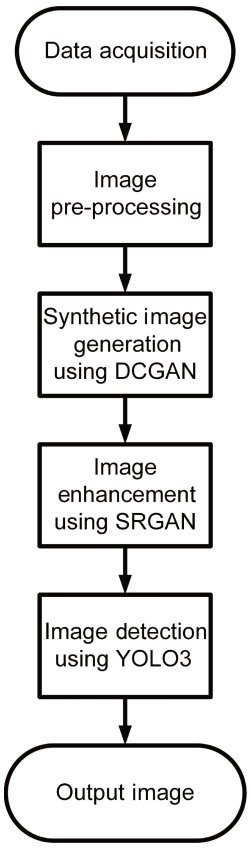

Fig. 5 Integrated data flow for innovative insulator monitoring system.

4.4 Model hyperparameters and training details

Table 2 summarizes the hyperparameters and training details for each component of the hybrid model. Careful selection of these values was made to achieve optimal performance.

4.5 Training procedures for each component

The training procedures for each component are outlined in Table 3, including data preprocessing, augmentation, and convergence criteria.

4.6 Evaluation of super-resolution image models

In this study, the authors used two advanced deep learning based super-resolution models for assessment

Table 2 Hybrid model hyperparameters and training details.

Component	Learning rate	Batch size	Activation function	Epochs
DCGAN	0.0002	64	LeakyReLU	200
YOLO3	0.001	16	LeakyReLU	150

Table 3 Training procedures for each component.

Component	Data pre-processing	Augmentation	Convergence criteria
SRGAN	Resize to 128×128	None	100 epochs
DCGAN	Resize to 128×128	Data augmentation	200 epochs
YOLO3	Resize to YOLO3 input	None	150 epochs

and comparison. The most common metric used to calculate a super-resolution model's outcome is the PSNR: peak signal to noise ratio. When assessing image reconstruction, this metric is used because it compares how close an image is by the mean square error to another and is expressed in decibels.

$$\mathrm{PSNR} = 10\log_{10}\left\{\frac{R^2}{\mathrm{MSE}}\right\} \tag{8}$$

where R is the maximum size that pixels will take in the image format used, which is normally 1 or 255. The maximum value is 255 in this instance. MSE is the mean square error which is to be compared between the two images. The resemblance between the two images is calculated by the structural similarity (SSIM). Furtherinformationonthisparametercanbefoundin Ref.[33]. The outcome is a value between −1 and 1, where, 1 indicates that the two images are identical. Table 4 shows the comparison of two super-resolution models with respect to PSNR (dB) and SSIM. From the table, it is clear that the SRGAN is performing better than SRCNN. The super-resolution image output for the suggested two models is seen in Fig. 6. Figures 7 and 8 display the loss curves obtained after the training and testing process for the generator and discriminator, respectively.

Table 4 Comparison of super-resolution models.

Parameter	SRCNN	SRGAN
PSNR (dB)	27.24	29.78
SSIM	0.814	0.872

4.7 Evaluation of DCGAN

The authors focused on creating synthetic insulator images with GAN for data augmentation to increase small datasets and improve efficiency on classification tasks using deep learning. The fake insulator images generated using DCGAN is illustrated in Fig. 9. However, traditional augmentation methods were also performed to demonstrate the efficacy of the proposed DCGAN. Operations such as movement, rotation, and brightness improvement were involved in the popular augmentation process we used in this work. As the training set and validation set of the deep neural network, we use the original images and the created images as seen in Fig. 9. Table 5 shows the experimental results. The average accuracy of recognition using training samples augmented by DCGAN is approximately 11% higher than that of the common method of augmentation. The results suggest that DCGAN is improving the accuracy of the classifier to some degree. The loss curves of generator and discriminator for 5000 epochs is shown in Fig. 10.

4.8 Performance comparison of object detecting models

The latest models for object detection include faster RCNN, YOLO2, and YOLO3. We employed these deep learning models to conduct experiments and evaluate their accuracy in our proposed work.

In our analysis, we considered three cases:

(1) The sole use of the object detection model (YOLO3). (2) Combining the super-resolution model (SRGAN) with the object detection model (YOLO3). (3) Implementing a hybrid model, which extends the second case. Here, we introduced a fake image

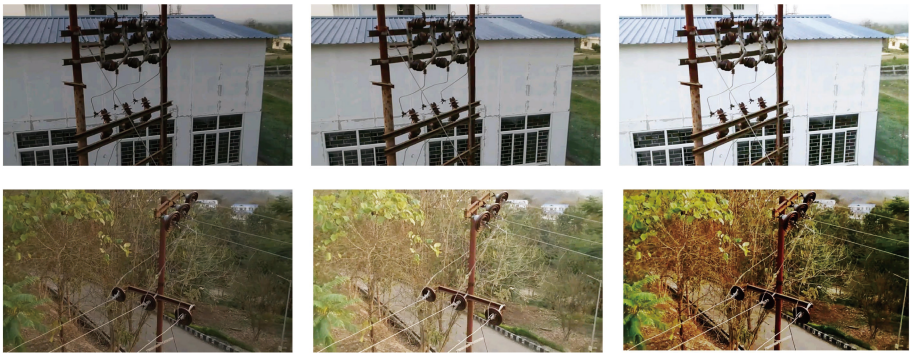

Fig. 6 From left to right direction: original image, SRCNN output, and SRGAN output.

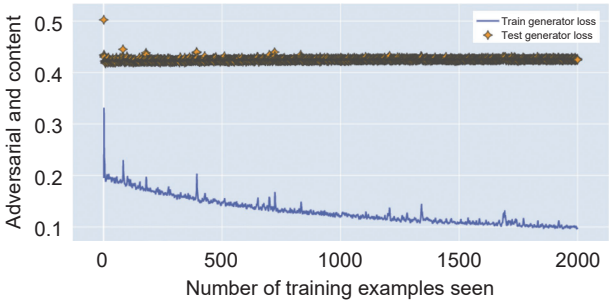

Fig. 7 Generator loss curves.

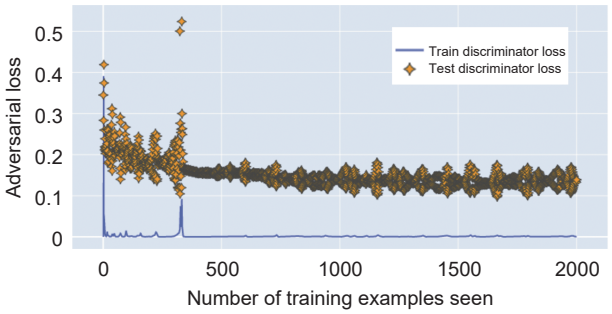

Fig. 8 Discriminator loss curves.

generator (DCGAN) between SRGAN and YOLO3 to address data imbalance issues.

As described above, we conducted a performance comparison of these three object detection models using our private insulator dataset. The results clearly indicate that YOLO3 outperforms the other two models in terms of sensitivity, specificity, and accuracy. Table 6 provides a comprehensive comparison of the latest object detection models, including a baseline model. The table also includes the average processing time per image (APTI) for each model, along with other standard metrics. These results underscore that YOLO3 not only offers the fastest performance but also achieves the highest accuracy among object detection models.

Following the implementation of the three cases mentioned above, we conducted a comparison based on standard deep learning parameters, including sensitivity, specificity, and accuracy, as presented in Table 7. Additionally, we extended our evaluation to include parameters such as localization accuracy, damage severity assessment, and false alarm rate. The table clearly demonstrates that our proposed hybrid model, SRGAN+DCGAN+YOLO3, exhibits significantly improved sensitivity, specificity, and accuracy compared to the other classifiers.

Furthermore, the SRGAN+DCGAN+YOLO3 model shows superior performance in localization accuracy, indicating a more precise identification of damage locations. In terms of damage severity assessment, this model demonstrates a higher capability in correctly determining the severity of insulator damage, which is crucial for practical maintenance and repair prioritization. The false alarm rate for the proposed model is also notably lower, suggesting a reduced likelihood of misidentifying healthy insulators as damaged. These additional parameters underscore the robustness of the SRGAN+DCGAN+YOLO3 model, making it a more reliable and effective tool for insulator damage detection in practical applications.

Finally, the results of our proposed hybrid approach

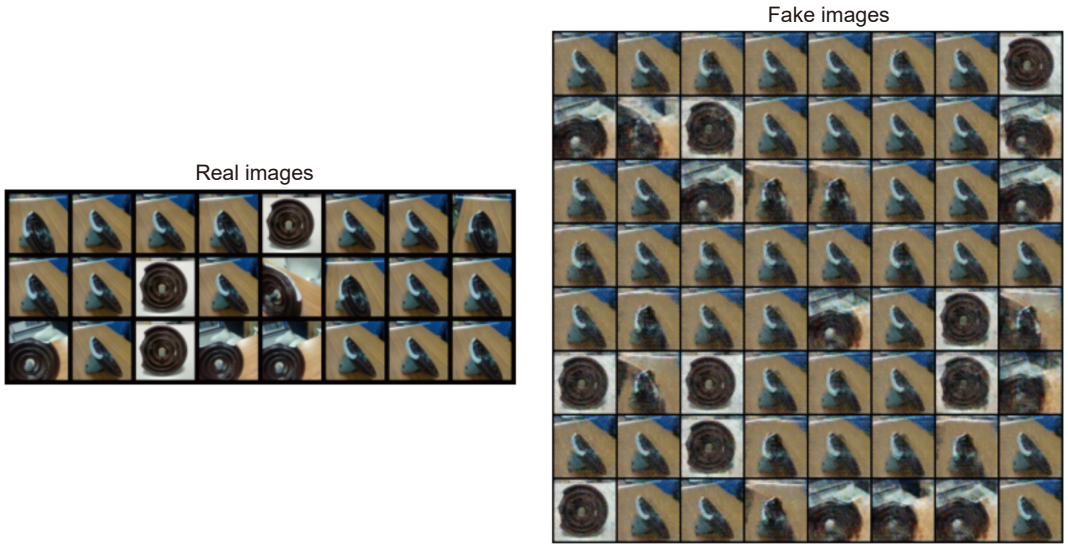

Fig. 9 Real image samples (left) and fake images (right).

Table 5 Comparison of augmentation models.

Model	No. of synthetic images	Accuracy
Standard methods	1000	75.3%
DCGAN	1000	86.5%

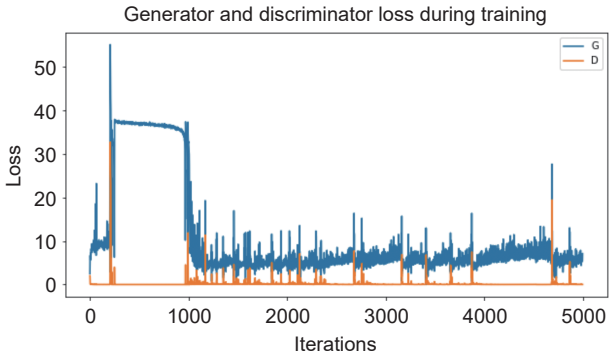

Fig. 10 Loss curves for generator (G) and discriminator (D) of DCGAN.

are depicted in Fig. 11, where you can observe the test images along with their corresponding output images, including bounding boxes, and class names.

5 Conclusion

In this study, we propose a hybrid model for identifying insulator damage in aerial images, aiming to overcome existing limitations in the field. Our approach integrates three crucial components to achieve this goal. Firstly, we employ image processing techniques in conjunction with SRGAN to detect low-resolution images and enhance them into high-resolution counterparts. This step is essential for improving the quality of the input data. Secondly, we address data imbalance challenges by incorporating an advanced DCGAN, which generates synthetic images. This helps create a more balanced dataset for training and enhances the model's ability to detect insulator damage effectively. Thirdly, we harness the power of deep learning through the YOLO3 model, which is applied to the surveillance images for insulator health detection. This deep learning-based approach enhances the accuracy and reliability of the detection process. Our extensive experimentation demonstrates the effectiveness of the proposed approach. Notably, the SRGAN outperforms SRCNN, achieving the highest peak signal-to-noise ratio. Furthermore, the hybrid solution, denoted as SRGAN+DCGAN+YOLO3, exhibits an impressive 95.6% detection accuracy, showcasing its superior performance compared to other implemented classifiers. Looking ahead, our future research endeavors will focus on expanding the

Table 6 Comparison of state-of-the-art object detection models.

Model	Sensitivity (%)	Specificity (%)	Accuracy (%)	APTI
Faster RCNN	84	81	88.6	0.72
YOLO2	92.2	83.6	89.3	0.54
YOLO3	92.4	91	90.5	0.33
Baseline model	88	76	82.5	0.65

Table 7 Comparison of implemented classifiers.

Model	Sensitivity	Specificity	Accuracy	Localization accuracy	Damage severity	False alarm rate
Only YOLO3	90%	81	86.9%	80%	82%	15%
SRGAN+YOLO3	96.6%	85%	92%	85%	87%	12%
SRGAN+DCGAN+YOLO3	98%	94%	**95.6%**	90%	92%	8%
Baseline model	88%	76%	82.5%	75%	78%	18%

Fig. 11 YOLO3 output images.

training dataset by incorporating more aerial samples. Additionally, we plan to explore further advancements in image enhancement techniques and automate annotations to maximize the potential of our proposed framework.

References

[1] S. Anjum, S. Jayaram, A. El-Hag, and A. N. Jahromi, Detection and classification of defects in ceramic insulators using RF antenna, *IEEE Trans. Dielect. Electr. Insul.*, vol. 24, no. 1, pp. 183–190, 2017.

[2] K. Marimuthu, S. Vynatheya, N. Vasudev, and P. Raja, Quality Analysis of Ceramic Insulators Under Electro Thermal Stresses, in *Proc. 2019 Int. Conf. on High Voltage Engineering and Technology* (*ICHVET*), Hyderabad, India, 2019, pp. 1–6.

[3] M. T. Gencoglu and M. Uyar, Prediction of flashover voltage of insulators using least squares support vector machines, *Expert Syst. Appl.*, vol. 36, no. 7, pp. 10789–10798, 2009.

[4] L. Yang, F. Zhang, and Y. Hao, Effects of structure and material of polluted insulators on the wetting characteristics, *IET Sci. Meas. Technol.*, vol. 13, no. 2, pp. 131–138, 2018.

[5] G. Montoya, I. Ramirez, and J. I. Montoya, Correlation among ESDD, NSDD and leakage current in distribution insulators, *IEE Proc., Gener. Transm. Distrib.*, vol. 151, no. 3, p. 334, 2004.

[6] G. H. Vaillancourt, J. P. Bellerive, M. St-Jean, and C. Jean, New live line tester for porcelain suspension insulators on high-voltage power lines, *IEEE Trans. Power Deliv.*, vol. 9, no. 1, pp. 208–219, 1994.

[7] K. L. Wong, Application of very-high-frequency (VHP) method to ceramic insulators, *IEEE Trans. Dielect. Electr. Insul.*, vol. 11, no. 6, pp. 1057–1064, 2004.

[8] X. Ouyang, Z. Jia, S. Yang, X. Shang, X. Wang, H. Chen, D. Zhou, and R. Liu, Influence of algae growth on the external insulation performance of HVDC insulators, *IEEE Trans. Dielect. Electr. Insul.*, vol. 25, no. 1, pp. 263–271, 2018.

[9] V. Murthy, K. Tarakanath, D. Mohanta, and S. Gupta, Insulator condition analysis for overhead distribution lines using combined wavelet support vector machine (SVM), *IEEE Trans. Dielect. Electr. Insul.*, vol. 17, no. 1, pp. 89–99, 2010.

[10] R. Li, H. Gu, B. Hu, and Z. She, Multi-feature fusion and damage identification of large generator stator insulation based on lamb wave detection and SVM method, *Sensors*, vol. 19, no. 17, pp. 3733, 2019.

[11] D. Mussina, A. Irmanova, P. K. Jamwal, and M. Bagheri, Multi-modal data fusion using deep neural network for condition monitoring of high voltage insulator, *IEEE Access*, vol. 8, pp. 184486–184496, 2020.

[12] D. T. Nguyen, T. N. Nguyen, H. Kim, and H. J. Lee, A high-throughput and power-efficient FPGA implementation of YOLO CNN for object detection, *IEEE Trans. VLSI Syst.*, vol. 27, no. 8, pp. 1861–1873, 2019.

[13] S. Khan, N. Islam, Z. Jan, I. Ud Din, and J. J. P. C. Rodrigues, A novel deep learning based framework for the detection and classification of breast cancer using transfer learning, *Pattern Recognit. Lett.*, vol. 125, pp. 1–6, 2019.

[14] X. Tao, D. Zhang, Z. Wang, X. Liu, H. Zhang, and D. Xu, Detection of power line insulator defects using aerial images analyzed with convolutional neural networks, *IEEE Trans. Syst. Man Cybern, Syst.*, vol. 50, no. 4, pp. 1486–1498, 2020.

[15] M. T. Gençoğlu and M. Cebeci, Investigation of pollution flashover on high voltage insulators using artificial neural network, *Expert Syst. Appl.*, vol. 36, no. 4, pp. 7338–7345, 2009.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet classification with deep convolutional neural networks, *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[17] L. Singh, A. Alam, K. V. Kumar, D. Kumar, P. Kumar, and Z. A. Jaffery, Design of thermal imaging-based health condition monitoring and early fault detection technique for porcelain insulators using machine learning, *Environ. Technol. Innov.*, vol. 24, p. 102000, 2021.

[18] S. F. Stefenon, L. O. Seman, N. F. Sopelsa Neto, L. H. Meyer, V. C. Mariani, and L. S. Coelho, Group method of data handling using Christiano–Fitzgerald random walk filter for insulator fault prediction, *Sensors*, vol. 23, no. 13, p. 6118, 2023.

[19] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Generative adversarial nets, in *Proc. 27th Int. Conf. Neural Information Processing Systems - Volume 2*, Montreal, Canada, 2014, pp. 2672–2680.

[20] Y. LeCun, Y. Bengio, and G. Hinton, Deep learning, *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[21] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Generative adversarial nets, in *Proc. 27th Int. Conf. Neural Information Processing Systems - Volume 2*, Montreal, Canada, 2014, pp. 2672–2680.

[22] H. Emami, M. M. Aliabadi, M. Dong, and R. B. Chinnam, SPA-GAN: Spatial attention GAN for image-to-image translation, *IEEE Trans. Multimedia*, vol. 23, pp. 391–401, 2021.

[23] L. Gao, D. Chen, Z. Zhao, J. Shao, and H. T. Shen, Lightweight dynamic conditional GAN with pyramid attention for text-to-image synthesis, *Pattern Recognit.*, vol. 110, p. 107384, 2021.

[24] Y. Yu, Z. Huang, F. Li, H. Zhang, and X. Le, Point Encoder GAN: A deep learning model for 3D point cloud inpainting, *Neurocomputing*, vol. 384, pp. 192–199, 2020.

[25] U. Babawuro, B. Zou, and B. Xu, High resolution satellite imagery rectification using Bi-linear interpolation method for geometric data extraction, in *Proc. Second Int. Conf. Intelligent System Design and Engineering Application*, Sanya, China, 2012.

[26] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, You only look once: Unified, real-time object detection, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (*CVPR*), Las Vegas, NV, USA, 2016.

[27] J. Redmon and A. Farhadi, YOLO9000: Better, faster,

stronger, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (*CVPR*). Honolulu, HI, USA, 2017.

[28] J. Redmon, and A. Farhadi, YOLO3: An incremental improvement, arXiv preprint arXiv:1804.02767, 2018.

[29] T. Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, Feature pyramid networks for object detection, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (*CVPR*), Honolulu, HI, USA, 2017.

[30] C. Dong, C. C. Loy, K. He, and X. Tang, Learning a deep convolutional network for image super-resolution, in *European Conf. Computer Vision*, Zurich, Switzerland, 2014.

[31] C. Ledig, L. Theis, F. Husz´ar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi, Photo-realistic single image super-resolution using a generative adversarial network, in *Proc. IEEE Conference on Computer Vision and Pattern Recognition 2017*, Honolulu, HI, USA, pp. 4681–4690, 2017.

[32] D. Sarkar and S. K. Gunturi, Wind turbine blade structural state evaluation by hybrid object detector relying on deep learning models, *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–4, 2020.

[33] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, Image quality assessment: From error visibility to structural similarity, *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

Ramakrishna Akella currently working as a principal, in Koneru Lakshmaiah Education Foundation, Hyderabad campus, is an accomplished electronics and communication engineer with a wealth of experience in academia, research, and industry. He is also a doctoral researcher in wireless communications and received the master degree in telecommunications & electronics engineering from Sheffield Hallam University, UK. He started his career as a design engineer at Sheffield Centre for Robotics and later worked as a telecom engineer in British Telecommunications. His areas of interest are wireless communication, device to device communication, machine learning, and deep learning.

Sravan Kumar Gunturi received the BTech degree from Sasi Institute of Technology and Engineering Tadepalligudem, India in electronics and communication engineering, 2009. He received the MTech degree in specialization in embedded systems from BVC Engineering College, India, in 2012 and received the PhD degree from Department of Electronics and Instrumentation Engineering, National Institute of Technology, India. Currently he is working as an associate professor in Koneru Lakshmaiah Education Foundation, India. His areas of interest are machine learning, deep learning, and the Internet of Things.

Dipu Sarkar received the BTech in electrical engineering in 2003 from University of Kalyani, India. He received the MTech degree in specialization of electrical power systems from University of Calcutta, India, in 2007, and the PhD degree from Department of Electrical Engineering, Bengal Engineering and Science University, India (presently known as Indian Institute of Engineering Science and Technology) in 2013. Currently he is working as an associate professor in Department of Electrical and Electronics Engineering, National Institute of Technology, India. His fields of interest are power systems operation and control, power systems stability, soft computational applications in power systems, and smart grids, grid integrated renewable energy, service restoration, and protective relay coordination.