Fault Detection in Power Distribution Systems Using Sensor Data and Hybrid YOLO with Adaptive Context Refinement

Luiza Scapinello Aquino , Luis Fernando Rodrigues Agottani , Laio Oriel Seman * , Viviana Cocco Mariani , Leandro dos Santos Coelho , Gabriel Villarrubia Gonzalez

Posted Date: 16 July 2025

doi: 10.20944/preprints202507.1307.v1

Keywords: transmission line's insulators; object detection; deep learning; computer vision; convolutional neural networks; YOLOv8; multi-scale analysis

 doi:10.20944/preprints202507.1307.v1

Article

Fault Detection in Power Distribution Systems Using Sensor Data and Hybrid YOLO with Adaptive Context Refinement

Luiza Scapinello Aquino[1], Luis Fernando Rodrigues Agottani[2], Laio Oriel Seman [3,*], Viviana Cocco Mariani [2,4], Leandro dos Santos Coelho [1,4] and Gabriel Villarrubia González [3]

[1] Graduate Program in Electrical Engineering, Federal University of Parana, Curitiba, PR, Brazil
[2] Graduate Program in Mechanical Engineering, Federal University of Parana, Curitiba, PR, Brazil
[3] Department of Automation and Systems Engineering, Federal University of Santa Catarina, Florianopolis, SC, Brazil
[4] Department of Electrical Engineering, Federal University of Parana, Curitiba, PR, Brazil
[3] Expert Systems and Applications Lab, Faculty of Science, University of Salamanca, Plaza de los Caídos s/n, 37008 Salamanca, Spain
* Correspondence: laioseman@gmail.com

Abstract

Ensuring the reliability of power transmission systems hinges on the accurate detection of defects in insulators, which are vulnerable to environmental degradation and mechanical stress. Traditional inspection methods are time-consuming and often ineffective, particularly in complex aerial environments. To address these challenges, this paper presents a hybrid fault detection framework that integrates the YOLOv8 object detection model with a novel Adaptive Context Refinement (ACR) mechanism. While YOLOv8 enables real-time detection, the ACR module enhances precision by adaptively leveraging multi-scale contextual information surrounding detected objects. This joint architecture improves the detection of subtle and small-scale insulator defects, especially under challenging imaging conditions. The proposed system is evaluated across 25 YOLO model variants using high-resolution datasets, including real-world inspection images from power distribution networks. Results show that ACR significantly improves mean Average Precision, particularly in lightweight models like YOLOv10n, which achieved a 22.9% performance gain. The findings demonstrate the effectiveness of contextual refinement in reducing false positives and enhancing defect classification, supporting practical deployment in automated aerial inspection systems.

Keywords: transmission line's insulators; object detection; deep learning; computer vision; convolutional neural networks; YOLOv8; multi-scale analysis

1. Introduction

Insulators are essential parts of high-voltage transmission networks because they provide electrical insulation and mechanical support to conductors [1]. These components, which provide electrical insulation between energized conductors and supporting structures, are continually exposed to adverse environmental conditions that accelerate their degradation over time [2]. Their effectiveness has a direct impact on the electricity grid's stability and safety. Insulators exposed to electrical, mechanical, and environmental stresses in outdoor applications are more likely to develop flaws, including leakage current, self-explosion, partial fractures, and pollution-induced electrical discharges [3].

In electrical power insulators, mechanical defects are frequently caused by manufacturing flaws and accumulated stress, whereas pollution-induced electrical discharges are caused by the accumulation of impurities that, when dissolved, produce a conductive layer on the surface of the insulator [4]. This conductive channel increases leakage current and can cause partial discharges, decreasing the insulator's dielectric strength and system reliability [5].

 doi:10.20944/preprints202507.1307.v1

These vulnerabilities constitute a serious danger to transmission continuity and the integrity of the power system, with insulator faults being the cause of transmission-related incidents [6]. As a result, routine inspections are necessary. Traditional inspection methods, such as field patrols and manned helicopter monitoring, are resource-intensive and ineffective [7]. Although unmanned aerial vehicles (UAVs) provide greater coverage, problems remain due to the complexity of the transmission infrastructure, differences in brightness, small fault sizes, and background interference in the image [8].

Thus, conventional inspection techniques for overhead power transmission lines are largely insufficient due to the considerable height and breadth of transmission towers [9]. Identification and examination of electrical insulators in aerial photographs with obstructed backgrounds provide significant challenges for autonomous inspections [10]. However, the reliability of the power transmission infrastructure depends on several factors, including the operational integrity of electrical insulators [11].

Equally important are the effects of voltages when power cables are placed in acidic environments Preduș et al. [12]. As power grids expand and age, the ability to accurately detect and classify insulator defects can prevent interruptions in power distribution service and also maintain distribution efficiency [13]. Stefenon et al. [14] combined the YOLO with DenseNet-161 and a Quasi-ProtoPNet structure to have a hybrid structure that outperforms standard insulator fault detection methods.

In the electrical sector, as in other areas, there is a gradual evolution in inspection methodologies, migrating from visual assessments, usually purely manual, to sophisticated technological approaches [15]. Freitas-Gutierres et al. [16] investigated the classification and positioning of partial discharges in insulators exposed to pollution, mud, and degradation by analyzing electric potential gradients through simulations in the finite element method. Subsequently, Mahapatra et al. [17] applied the projected gradient descent and the fast gradient signal approach for insulator defect classification problems.

Despite these advances, detecting faults in insulators remains challenging because of the diversity of defect manifestations, the variability in installation configurations, and mainly due to the positioning, often remote or elevated, of these components [18]. This complexity requires inspection systems capable of broad coverage and detailed analysis [19]. Traditional object detection models like You Only Look Once (YOLO), while effective in general applications, struggle with these specific issues due to their limited use of contextual information and fixed anchor box assumptions.

This paper introduces an enhanced approach to insulator fault detection by integrating Adaptive Context Refinement (ACR) with the YOLO-v8 detection framework. Although YOLO-v8 provides efficient initial detection capabilities, the ACR enhancement leverages surrounding visual context to improve detection precision and fault classification accuracy. The key innovation lies in the adaptive selection of contextual regions based on the characteristics of the insulator, enabling more robust performance under varying imaging conditions.

The main contributions of this study can be summarized as follows:

- Proposes a novel ACR mechanism that enhances YOLOv8-based object detection by incorporating contextual information surrounding insulator defects, improving classification accuracy and localization precision in complex scenarios.
- Designs an adaptive multi-scale context extraction strategy that dynamically selects context region sizes based on the object's relative area, enabling better detection of small or partially occluded faults in power transmission insulators.
- Integrates a lightweight attention-based refinement network, including spatial and channel attention modules, which refines the bounding boxes and recalibrates confidence scores without significantly increasing computational cost.
- Conducts a comprehensive evaluation across 25 YOLO model variants (YOLOv8 through YOLOv12), demonstrating consistent improvements in mean Average Precision (mAP), particularly for lower-capacity models with limited baseline accuracy.

 doi:10.20944/preprints202507.1307.v1

- Validates the approach using real-world UAV datasets containing high-resolution images of insulators under varying environmental and fault conditions, proving its robustness and practical applicability to real-time inspection in power systems.

The remainder of the paper is organized as follows: Section 2 reviews related works on the application of artificial intelligence methods for power grid fault identification. Section 3 describes the proposed methodology, followed by the experimental setup in Section 4. Section 5 presents the results and discussion. Finally, Section 6 provides the conclusions.

2. Related Works

Recent developments in algorithms for feature extraction and classification of anomalies based on data or computer vision have created new possibilities for automating the detection of anomalies in power grid components [20]. These algorithms are typically categorized into three categories: machine learning [21], deep learning (DL) [22], and ensemble learning models [23]. The most popular DL models are Residual Networks, Visual Geometry Group, YOLO [24], and hybrid models [25]. Methods such as Faster Region Based Convolutional Neural Networks (Faster R-CNN) [26] also offer high accuracy but suffer from slow inference and complex training, and methods like YOLO versions prioritize speed, efficiency, and competitive accuracy.

One challenge in identifying faults in power line insulators is data availability. Song et al. [27] investigated different techniques for data augmentation and their effects. The authors observed that through Gaussian blurring, scaling, and rotation, data augmentation plays an important role in the accuracy of recognition models. Han et al. [28] applied an enhanced SIoU loss function to obtain regression results and utilized an ECA module between the backbone and the feature fusion layer of the YOLOX-s model, achieving an mAP of 97.18%. Wang et al. [29] developed a new network incorporating the region proposal network for feature extraction, allowing better detection of small defects in insulators with an accuracy of 98.38% but with prohibitive computational time.

Additionally, as emphasized by Shuang et al. [30], the integration of a feature enhancement module in conjunction with an attention mechanism, applied within the Faster R-CNN framework, demonstrated improved representation of object regions, resulting in an mAP of 96.6%. However, that approach exhibited relatively slow computational performance. Zhou et al. [31] used the Mask R-CNN model to detect defects in insulators using infrared images, achieving an accuracy of up to 98%, and a number of frames per second of 5.75. in [32], a Faster R-CNN was applied to accurately identify eight categories of defects in transmission lines using complex backgrounds and different illuminations.

In terms of this, researchers have used several methods to detect power line insulator defects effectively. In the study [33], super-resolution CNN was applied to reconstruct the blurred image, and the inspection image was based on YOLO-V3, increasing the accuracy 1% and 3% higher than the Faster R-CNN and Shot Multibox Detector, respectively. Similarly, Han et al. [34] YOLO-v3 was trained and used to detect the insulator faults with an average execution time of about 30 ms, demonstrating its potential to be adopted for online detection. Costa and Cortes [35] three versions of the YOLO-V3 were compared against the state-of-the-art CNN, with the best model reaching a precision of 98%.

Nonetheless, Qiu et al. [36] proposed an enhanced YOLOv4-based model for insulator defect detection, achieving a detection accuracy of approximately 93.81% with an inference speed of 53 frames per second. Furthermore, the application of image preprocessing techniques increased the accuracy to 97.26%. Liquan et al. [37] proposed a faster insulator fault detection method based on YOLO-v5s, featuring a faster detection speed and acceptable detection accuracy. Zhang et al. [38] introduced the Ghost module into the backbone and neck of YOLO-v5 to reduce the parameters and model size in order to improve the performance of unmanned aerial vehicles (UAVs) with improved performance of YOLO-v5. Zheng et al. [39] proposed an improved YOLOv7 model for small object detection on complex backgrounds, achieving an average accuracy of 93.8%, and outperforming Faster R-CNN, YOLOv7, and YOLOv5s models by 7.6%, 3.7%, and 4%, respectively. On the other hand, Chen et al. [40]

 doi:10.20944/preprints202507.1307.v1

designed a new ID-YOLOv7 that achieved a mAP of 85.7%, representing a 7.2% increase over the original YOLOv7.

For instance, Zhang et al. [41] the improved YOLO-v7 model achieved of 95% average detection accuracy (mAP), recall, and precision in two datasets, TISLTR and FISLTR, to detect isolator targets. Gong et al. [42] proposed an insulator fault diagnosis method based on YOLO-v8-DCP multi-mechanism optimization by improving multi-scale feature extraction and multi-level feature fusion, indicating that the model achieves an accuracy of 97.7% in insulator fault diagnosis. in [43], another improved YOLO-v8 model was proposed for aerial photography scenarios by UAV, called UAV-YOLO-v8, with fewer parameters and 7.7% higher average detection accuracy than the base model. Tao et al. [44] proposed a model based on YOLO-v8, achieving a mAP of 98.8% for situations with few types of defects.

He et al. [45] proposed an enhanced object detection framework, named MFI-YOLO, to accurately identify and classify multiple insulator fault types in aerial images, particularly under complex backgrounds and varying fault scales. They addressed the limitations of traditional detection models by improving the YOLOv8 architecture. Specifically, they replaced the original C2F network with a more efficient structure that combines GhostNet and multi-scale asymmetric convolution (MSA-GhostBlock), allowing better feature extraction in visually complex environments. Additionally, they introduced ResPANet, a residual-based multi-scale feature fusion module, to replace the conventional PANet and enhance detection performance in multi-target scenarios. To train and validate the proposed model, the authors constructed a dedicated dataset containing four categories of insulators: normal, self-explosive, damaged, and flashover. Through extensive experiments, they demonstrated that the proposed MFI-YOLO significantly improves detection accuracy, raising the mAP from 89.2% to 93.9%. Their model showed high effectiveness, particularly in recognizing subtle and hard-to-detect insulator faults, highlighting its potential for practical applications in power line inspection.

In general, hybrid methods have gained prominence in these applications, such as MCI-GLA proposed by Wang et al. [46], A2MADA-YOLO presented by Li et al. [47], IN-YOLO proposed by Sadykova et al. [48], CACS-YOLO of [49], IL-YOLO proposed by Zhang et al. [50], ID-YOLO presented by Zhang et al. [51], GC-YOLO of Ding et al. [52], LiteYOLO-ID presented by Li et al. [53], HRD-YOLOX proposed by Li et al. [54], DRR-YOLO presented by Hu et al. [55], LA-YOLO proposed by Liu and Jiang [56], and LDDFSF-YOLO11 presented by Shen et al. [57].

Zhang et al. [58] proposed an improved YOLO-v9 model, achieving an mAP of 96.6% but with limitations in complex backgrounds, did not consider multiple types of defects in insulators, and had higher computational complexity than YOLO-v9. Zhao et al. [59] proposed an improved approach to defect detection based on a lightweight neural network derived from the YOLOv11n architecture, incorporating multidimensional dynamic convolutions for feature extraction. Results demonstrate that the proposed method outperforms YOLO-v8 and YOLO-v10. Recently, Stefenon et al. [60] proposed an optimized ensemble of weighted box fusion (WBF)-based deep learning (OEDL) models, named OEDL-WBF, to improve fault detection in power grid insulators. With an average accuracy of 0.9841, the proposed OEDL-WBF outperformed YOLO-v8 to YOLO-v12 in a benchmarking test.

Despite these advances in the area of insulator detection in power transmission networks, attention mechanisms based on maximum and average pooling still face limitations, as both tend to discard important spatial or texture information, which is detrimental to the detection of small or detailed targets [61]. Object detection models, particularly designed for efficiency and accuracy in real-time applications, show promise in helping utilities monitor and maintain transmission lines. However, standard detection frameworks often struggle to detect distinctive visual patterns in insulator defects, which can be subtle and location-dependent.

The main challenge lies in distinguishing normal patterns from those indicative of deterioration or damage to insulators [62]. Environmental factors such as lighting, weather effects, and accumulated debris make classification difficult. In addition, the structure of the insulators, which consists of multi-

 doi:10.20944/preprints202507.1307.v1

ple disc- or pin-type elements, requires detection systems to consider both the individual components and the collective arrangement [63].

3. Methodology

Computer vision techniques, particularly deep learning-based object detection, offer promising solutions to automate insulator fault detection. Among these approaches, You Only Look Once (YOLO) has emerged as an efficient framework capable of real-time detection with competitive accuracy. In this study, an approach is proposed that enhances YOLO with ACR to improve insulator fault detection precision.

3.1. YOLO Detection Framework

YOLO operates as a single-stage detector that processes the entire image in a single forward pass of the network, enabling real-time detection capabilities (see Figure 1. Given an input image $I \in \mathbb{R}^{H \times W \times 3}$ with height H, width W, and 3 color channels, YOLO divides the image into a regular grid of $S \times S$ cells. Each cell is responsible for detecting objects whose center falls within its boundaries. This grid-based approach can be represented as $G = g_{i,j} | i \in 1, 2, ..., S, j \in 1, 2, ..., S$, where $g_{i,j}$ represents the cell at position (i, j) in the grid.

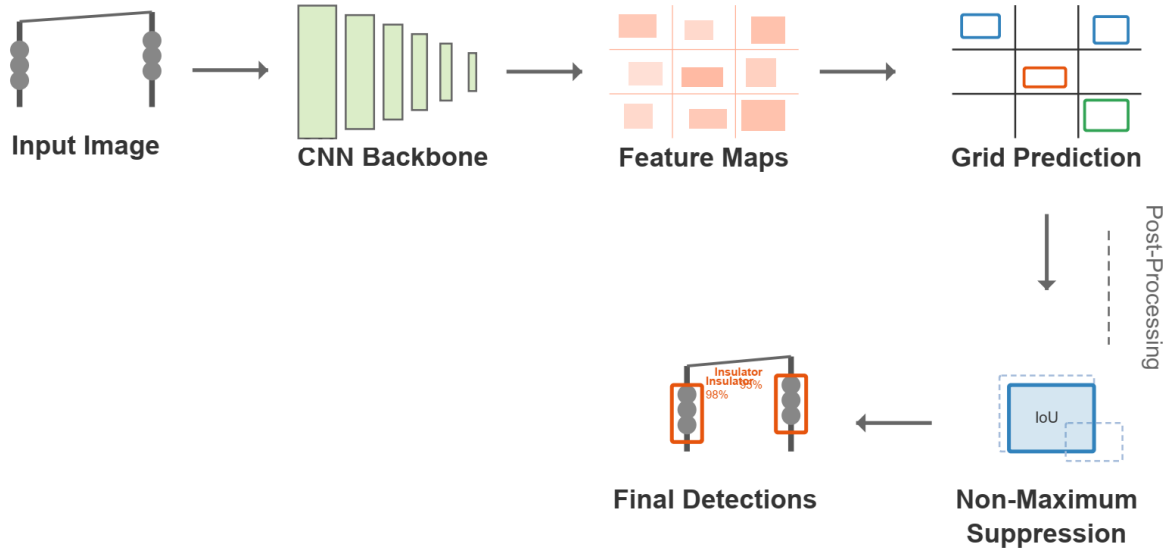

Figure 1. Workflow of YOLO approach.

The YOLO architecture consists of a backbone network (typically a convolutional neural network like DarkNet) followed by detection heads. The backbone extracts hierarchical features from the input image: $F = \Phi(I)$, where Φ represents the backbone network and F denotes the extracted feature maps. These feature maps are then processed by the detection heads to predict bounding boxes, objectness scores, and class probabilities for each grid cell: $P = \Psi(F)$, where Ψ represents the detection heads and $P \in \mathbb{R}^{S \times S \times B \times (5+C)}$ is the prediction tensor. Here, B is the number of bounding boxes predicted per grid cell, and C is the number of classes [64].

For each grid cell, YOLO predicts B bounding boxes. Each bounding box prediction consists of center coordinates (b_x, b_y) relative to the grid cell, width and height (b_w, b_h) relative to the image dimensions, objectness score p_o indicating the probability of an object being present, and class probabilities p_c for each of the C classes [65]. The network outputs are transformed to obtain the final bounding box parameters:

$$b_x = \sigma(t_x) + g_x \; b_y \qquad = \sigma(t_y) + g_y \; b_w = p_w e^{t_w} \; b_h \qquad = p_h e^{t_h} \tag{1}$$

 doi:10.20944/preprints202507.1307.v1

where (t_x, t_y, t_w, t_h) are the raw network outputs, σ is the sigmoid function that constrains the center coordinates to be within the corresponding grid cell, (g_x, g_y) are the coordinates of the top-left corner of the grid cell, and (p_w, p_h) are the width and height of predefined anchor boxes, which serve as prior information about typical object shapes [66]. The anchor boxes are determined through statistical analysis of the training dataset, typically using clustering algorithms like k-means on the dimensions of ground truth boxes:

$$(p_w^k, p_h^k)k = 1^B = \text{k-means}((w_i^{gt}, h_i^{gt})i = 1^N) \tag{2}$$

where (w_i^{gt}, h_i^{gt}) are the width and height of ground truth boxes, and N is the number of ground truth boxes in the training set.

The confidence score for each bounding box is computed as $s = \sigma(t_o) \cdot \text{IoU}(b, b^{gt})$, where t_o is the raw objectness score output by the network, $\sigma(t_o)$ represents the probability of an object being present, and $\text{IoU}(b, b^{gt})$ is the Intersection over Union between the predicted box b and the ground truth box b^{gt}. The IoU is calculated as:

$$\text{IoU}(b, b^{gt}) = \frac{\text{Area}(b \cap b^{gt})}{\text{Area}(b \cup b^{gt})} \tag{3}$$

where $\cap$ represents the intersection and $\cup$ represents the union of the two boxes.

The class prediction for each bounding box is determined by $P(c = k|\text{Object}) = \sigma(t_k)$, where t_k is the network output for class k, and $\sigma(t_k)$ represents the conditional probability of the object belonging to class k given that an object exists. The final class prediction is

$$c^* = \underset{k \in 1,2,\ldots,C}{\text{argmax}}, \sigma(t_k). \tag{4}$$

The YOLO loss function is a multi-part objective that balances localization accuracy, confidence prediction, and classification performance:

$$\mathcal{L} = \mathcal{L}loc + \mathcal{L}obj + \mathcal{L}noobj + \mathcal{L}class \tag{5}$$

The localization loss penalizes errors in bounding box coordinates and dimensions:

$$\mathcal{L}loc = \lambda coord \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} \left[(t_x^i - \hat{t}_x^i)^2 + (t_y^i - \hat{t}_y^i)^2 + (t_w^i - \hat{t}_w^i)^2 + (t_h^i - \hat{t}_h^i)^2 \right] \tag{6}$$

where λ_{coord} is a weighting factor that increases the importance of localization errors, $\mathbb{1}_{ij}^{obj}$ is an indicator function that equals 1 if the j-th bounding box predictor in cell i is responsible for detecting an object, and 0 otherwise, and $(t_x^i, t_y^i, t_w^i, t_h^i)$ and $(\hat{t}_x^i, \hat{t}_y^i, \hat{t}_w^i, \hat{t}_h^i)$ are the predicted and target box parameters, respectively.

The objectness loss penalizes errors in predicting object presence:

$$\mathcal{L}obj = \sum i = 0^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} \left[\sigma(t_o^i) - \hat{\sigma}(t_o^i) \right]^2 \tag{7}$$

where $\sigma(t_o^i)$ is the predicted objectness score and $\hat{\sigma}(t_o^i)$ is the target objectness score, typically set to the IoU between the predicted and ground truth boxes.

The no-object loss penalizes false positive predictions:

$$\mathcal{L}noobj = \lambda noobj \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{noobj} \left[\sigma(t_o^i) - 0 \right]^2 \tag{8}$$

 doi:10.20944/preprints202507.1307.v1

where λ_{noobj} is a weighting factor that typically reduces the influence of background regions and $\mathbb{1}_{ij}^{noobj}$ is an indicator function that equals 1 if the j-th bounding box predictor in cell i is not responsible for detecting any object, and 0 otherwise.

The classification loss penalizes errors in class prediction:

$$\mathcal{L}class = \sum i = 0^{S^2} \mathbb{1}i^{obj} \sum c = 1^{C} [p_i(c) - \hat{p}_i(c)]^2 \tag{9}$$

where $\mathbb{1}_i^{obj}$ is an indicator function that equals 1 if an object appears in cell i and 0 otherwise, $p_i(c)$ is the predicted probability of class c in cell i, and $\hat{p}_i(c)$ is the target probability (1 for the correct class, 0 for others).

After obtaining the raw predictions from the network, YOLO applies non-maximum suppression (NMS) to remove redundant detections. The NMS algorithm sorts all detections by confidence score in descending order, then for each class, selects the detection with the highest confidence score and removes all other detections of the same class that have IoU greater than a threshold θ_{NMS} with the selected detection. This process repeats until no detections remain above the confidence threshold θ_{conf}. The final set of detections after NMS is given by $\mathcal{D}(I) = (b_j, c_j, s_j)_{j=1}^{M}$, where M is the number of retained detections, $b_j = (x_1, y_1, x_2, y_2) \in \mathbb{R}^4$ denotes the bounding box coordinates (top-left and bottom-right corners), $c_j \in 1, 2, ..., C$ represents the predicted class, and $s_j \in [0, 1]$ indicates the confidence score.

Despite YOLO's effectiveness in general object detection tasks, several limitations arise when applying it to insulator fault detection. YOLO processes each grid cell relatively independently, with limited ability to incorporate broader visual context that may be important for distinguishing subtle insulator defects from normal variations. The predefined anchor boxes may not optimally capture the varying shapes and sizes of insulator components and their associated defects, particularly for elongated insulator strings with small defects [67].

The discretization of the image into a grid introduces inherent constraints on localization precision, which is particularly problematic for small defects on insulators that may span only a fraction of a grid cell. Environmental factors such as lighting, weathering, and partial occlusion can create visual ambiguities that standard classification approaches struggle to resolve without contextual information. Additionally, insulator defects can manifest at various scales, from microscopic cracks to large-scale damage, requiring multi-scale analysis capabilities that go beyond YOLO's standard feature pyramid approach [68].

To address these limitations, it was proposed to enhance the YOLO framework with ACR, a technique that leverages surrounding visual context to improve detection precision and fault classification accuracy. The following section provides a detailed description of ACR methodology and how it integrates with the YOLO detection framework.

3.2. Adaptive Context Refinement

Let $I \in \mathbb{R}^{H \times W \times 3}$ represent an input image with height H, width W, and 3 color channels. Given a base object detector $\mathcal{D}$ which outputs a set of detections $\{(b_i, c_i, s_i)\}_{i=1}^{N}$, goal is to refine these detections using contextual information. Each detection consists of a bounding box $b_i = (x_1, y_1, x_2, y_2) \in \mathbb{R}^4$, a class label $c_i \in \{1, 2, ..., C\}$, and a confidence score $s_i \in [0, 1]$.

Employ Adaptive Context Refinement, in short ACR, a technique that enhances detection by incorporating contextual information from regions surrounding the initial detections (see Figure 2. The refinement process can be formulated as a function $\mathcal{R}$ that maps the initial detections to refined detections:

$$\{(b_i', c_i, s_i')\}_{i=1}^{N} = \mathcal{R}(I, \{(b_i, c_i, s_i)\}_{i=1}^{N}) \tag{10}$$

where b_i' and s_i' represent the refined bounding box coordinates and confidence scores, respectively.

 doi:10.20944/preprints202507.1307.v1

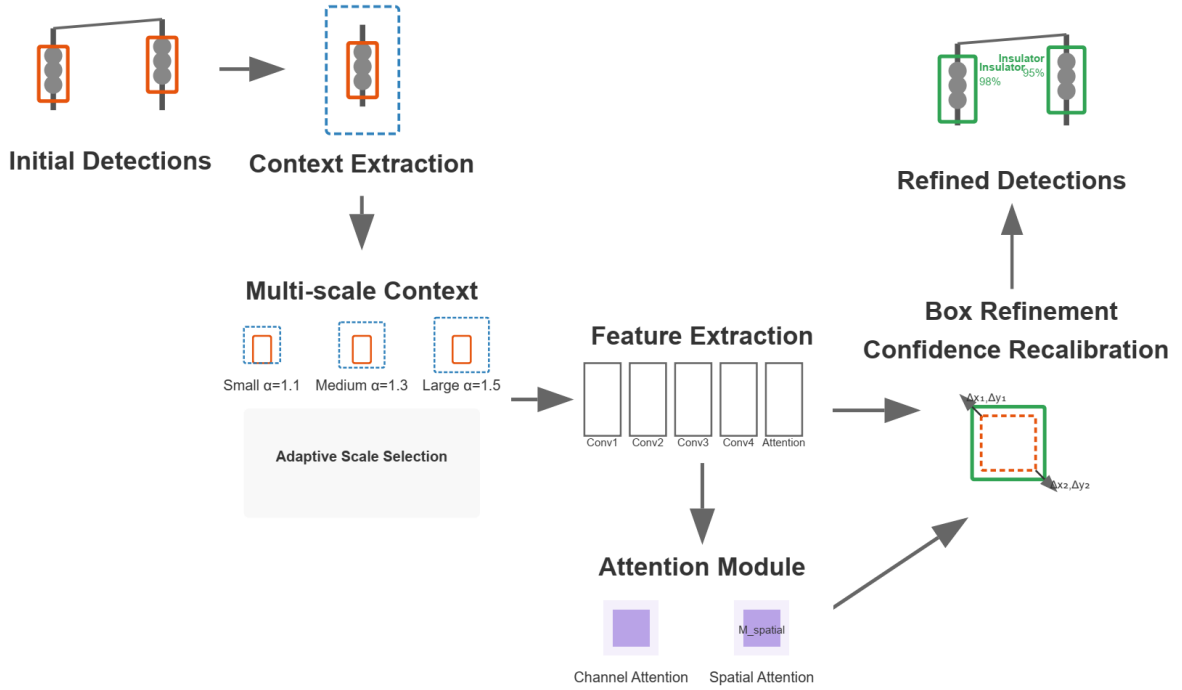

Figure 2. Adaptive Context Refinement.

3.3. Context Extraction and Neural Architecture

For each detection with confidence exceeding a threshold τ ($s_i > \tau$), extract multiple context regions at varying scales. For a detection box b_i, the context region at scale α is computed as:

$$b_i^\alpha = (c_x - \alpha\frac{w}{2}, c_y - \alpha\frac{h}{2}, c_x + \alpha\frac{w}{2}, c_y + \alpha\frac{h}{2}) \tag{11}$$

where the center coordinates (c_x, c_y) and dimensions (w, h) are derived from b_i:

$$c_x = \frac{x_1 + x_2}{2}, \quad c_y = \frac{y_1 + y_2}{2}, \quad w = x_2 - x_1, \quad h = y_2 - y_1 \tag{12}$$

The scale factor α is adaptively selected based on the relative size of the object:

$$\alpha = \begin{cases} \alpha_{\text{large}} & \text{if } \frac{wh}{HW} > \sigma_{\text{large}} \\ \alpha_{\text{medium}} & \text{if } \sigma_{\text{small}} < \frac{wh}{HW} \leq \sigma_{\text{large}} \\ \alpha_{\text{small}} & \text{if } \frac{wh}{HW} \leq \sigma_{\text{small}} \end{cases} \tag{13}$$

where $\sigma_{\text{small}} = 0.1$ and $\sigma_{\text{large}} = 0.5$ are thresholds, and $\alpha_{\text{small}} = 1.5$, $\alpha_{\text{medium}} = 1.3$, and $\alpha_{\text{large}} = 1.1$ are the context scales.

The convolutional feature extraction pathway consists of four layers with increasing receptive fields:

$$f_1 = \text{ReLU}(\text{BN}(\text{Conv}_{3\times3}(I_{b_i^\alpha}^{\text{resized}}, 16))) \tag{14}$$

$$f_2 = \text{ReLU}(\text{BN}(\text{Conv}_{3\times3}(f_1, 32))) \tag{15}$$

$$f_3 = \text{ReLU}(\text{BN}(\text{Conv}_{3\times3}(f_2, 64))) \tag{16}$$

$$f_4 = \text{ReLU}(\text{BN}(\text{Conv}_{3\times3}(f_3, 128))) \tag{17}$$

where $\text{Conv}_{3\times3}(X, n)$ denotes a 2D convolution with n filters of size 3×3 and stride 2, applied to input X, and BN denotes batch normalization.

 doi:10.20944/preprints202507.1307.v1

3.4. Spatial Attention and Feature Fusion

To focus on relevant spatial regions, incorporate a spatial attention mechanism:

$$f_{\text{avg}} = \text{AvgPool}_{\text{channel}}(f_4) \tag{18}$$
$$f_{\max} = \text{MaxPool}_{\text{channel}}(f_4) \tag{19}$$
$$f_{\text{concat}} = \text{Concat}(f_{\text{avg}}, f_{\max}) \tag{20}$$
$$M_{\text{spatial}} = \sigma(\text{Conv}_{7\times7}(f_{\text{concat}}, 1)) \tag{21}$$
$$f_{\text{att}} = f_4 \otimes M_{\text{spatial}} \tag{22}$$

where σ is the sigmoid activation function, and $\otimes$ denotes element-wise multiplication with channel-wise broadcasting. The spatial attention module produces an attention map M_{spatial} that highlights important regions in the feature maps.

Global average pooling is applied to obtain a fixed-length feature vector:

$$f_{\text{global}} = \text{GlobalAvgPool}(f_{\text{att}}) \in \mathbb{R}^{128} \tag{23}$$

The normalized bounding box coordinates within the context region are processed through a fully connected layer:

$$f_{\text{box}} = \text{ReLU}(\text{FC}(\hat{b}_i, 16)) \in \mathbb{R}^{16} \tag{24}$$

The image and box features are fused and processed to predict the refinement:

$$f_{\text{fused}} = \text{ReLU}(\text{FC}(\text{Concat}(f_{\text{global}}, f_{\text{box}}), 64)) \tag{25}$$
$$\Delta\hat{b}_i = \tanh(\text{FC}(f_{\text{fused}}, 4)) \cdot \delta \tag{26}$$

where $\delta = 0.1$ is a scaling factor to constrain the magnitude of refinements.

3.5. Box Refinement and Confidence Recalibration

The predicted refinement $\Delta\hat{b}_i$ is denormalized to the context region coordinates and applied to the original box:

$$\Delta b_i^{\alpha} = \left(\Delta\hat{b}_i[0] \cdot w^{\alpha}, \Delta\hat{b}_i[1] \cdot h^{\alpha}, \Delta\hat{b}_i[2] \cdot w^{\alpha}, \Delta\hat{b}_i[3] \cdot h^{\alpha}\right) \tag{27}$$
$$b_i' = (x_1 + \Delta b_i^{\alpha}[0], y_1 + \Delta b_i^{\alpha}[1], x_2 + \Delta b_i^{\alpha}[2], y_2 + \Delta b_i^{\alpha}[3]) \tag{28}$$

The final refined box is constrained to the image boundaries:

$$b_i' = \left(\max(0, b_i'[0]), \max(0, b_i'[1]), \min(W, b_i'[2]), \min(H, b_i'[3])\right) \tag{29}$$

The confidence score is recalibrated based on the IoU between the original and refined boxes:

$$s_i' = s_i \cdot \gamma(\text{IoU}(b_i, b_i')) \tag{30}$$

where γ is a piecewise function:

$$\gamma(\text{IoU}) = \begin{cases} 1.05 & \text{if IoU} > 0.9 \\ 1.02 & \text{if } 0.7 < \text{IoU} \leq 0.9 \\ 0.98 & \text{if } 0.5 < \text{IoU} \leq 0.7 \\ 0.90 & \text{if IoU} \leq 0.5 \end{cases} \tag{31}$$

 doi:10.20944/preprints202507.1307.v1

3.6. Training Methodology

The refinement network is trained using examples generated by matching base detector predictions with ground truth boxes. The training dataset consists of triplets $(I^{\text{resized}}_{b_i^{\alpha}}, \hat{b}_i, \hat{b}_i^{\text{gt}})$, where $\hat{b}_i^{\text{gt}}$ represents the normalized ground truth box coordinates in the context region. The network is trained to minimize the smooth L1 loss between the predicted refinement and the target refinement:

$$\mathcal{L} = \text{SmoothL1}(\Delta\hat{b}_i, \hat{b}_i^{\text{gt}} - \hat{b}_i) \tag{32}$$

The network is optimized using Adam with learning rate $\eta = 5 \times 10^{-4}$, weight decay $\lambda = 10^{-5}$, batch size $B = 32$, and trained for $E = 50$ epochs. A reduce-on-plateau scheduler with patience of 5 epochs and reduction factor of 0.5 is employed for learning rate adaptation.

During inference, multiple context scales are evaluated for each detection, and the refinement with the highest recalibrated confidence score is selected:

$$\alpha^* = \underset{\alpha \in \{\alpha_{\text{small}}, \alpha_{\text{medium}}, \alpha_{\text{large}}\}}{\arg\max} s_i'^{\alpha} \tag{33}$$

The final refined detection is $(b_i'^{\alpha^*}, c_i, s_i'^{\alpha^*})$. The computational complexity of the refinement network is $\mathcal{O}(K \cdot H_r W_r)$ for K context scales. For efficiency, apply refinement only to detections with confidence above threshold $\tau = 0.3$. The memory footprint of approximately 2.5M parameters makes the refinement network lightweight and suitable for deployment alongside the base detection model.

4. Experimental Setup

The algorithm implementations were developed in Python. For object detection tasks, a maximum limit of 100 training epochs was set, adopting the early stopping criterion if there was no improvement in performance over 10 consecutive epochs. In the context of synthetic image generation through generative models, up to 1,000 training cycles were considered. Performance evaluation was performed based on the number of epochs, with the aim of identifying the point at which the model reaches its best performance.

4.1. Performance Evaluation Metrics

To evaluate model performance, the metrics of precision, recall, F1-score, and mAP were adopted [60]. For object detection tasks, two Intersection over Union (IoU) thresholds were used: a fixed threshold of 0.5 (mAP@[0.5]) and a range from 0.5 to 0.95 in increments of 0.05 (mAP@[0.5:0.95]), determined as follows:

$$\text{precision} = \frac{tp}{tp + fp} \tag{34}$$

$$\text{recall} = \frac{tp}{tp + fn} \tag{35}$$

$$\text{F1-score} = \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}} \tag{36}$$

$$\text{mAP} = \frac{1}{n} \sum_{k=1}^{n} \left(\sum_{\eta} (\text{recall}_{\eta} - \text{recall}_{\eta-1}) \text{precision}_{\eta} \right)_k \tag{37}$$

where tp is the true positive, fp is the false positive, and fn is the false negative, η is the threshold associated with the n^{th} level, and k is the class label corresponding to that level among the n possible classes.

 doi:10.20944/preprints202507.1307.v1

4.2. Dataset

The analysis conducted in this study employed a dataset comprising high-resolution images of insulator strings, categorized into three sub-classes based on their condition: flashover-damaged insulator shells, broken insulator shells, and intact insulator shells. These images, originally captured during inspections of high-voltage transmission lines, are available through the repository developed by Lewis and Kulkarni [69].

The dataset includes insulator chains photographed under various conditions, including different angles and distances, insulators with visible structural damage, and surface degradation caused by flashover events. All images were acquired using high-resolution digital single-lens reflex cameras during transmission line inspections carried out under favorable weather conditions. Preprocessing steps included converting the original JSON-based annotations into YOLO-compatible files and resizing all images to 640 × 640 pixels, following the input specifications of the employed models.

Details regarding the number of annotated samples can be seen in [60]. For generalization, the proposed model is also applied to the dataset presented in [70], which contains 240 images of inspections recorded in an electrical power distribution network located in southern Brazil containing 13.8 kV. The set of images is divided into half-damaged and half-normal component conditions.

Figure 3 presents examples of insulators considered in this study. Figure 3A) shows an insulator chain having insulators in different conditions (we highlighted in red a broken insulator and in green an insulator in good condition) [69]. Figure 3b) shows an insulator with contamination on its surface and in the structure that supports the insulator [70].

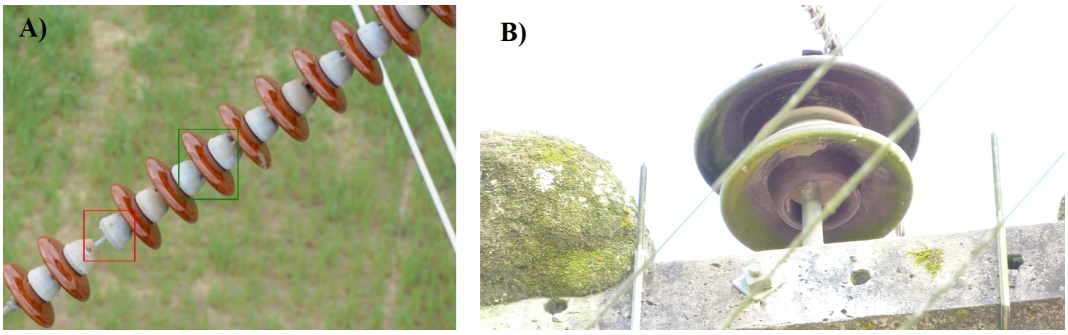

Figure 3. Insulator image samples: A) insulator chain having insulators in different conditions; B) contaminated insulator.

5. Results and Discussion

In this section, the results and discussions related to the method proposed in this paper are presented.

5.1. Training Methodology

The experimental implementation for training the proposed hybrid YOLO with ACR utilizes a systematic approach designed to evaluate model performance across different YOLO architectures. The training framework employs YOLO versions 8 through 10, applying various model capacities for each version to analyze the impact of model scale on fault detection accuracy. The training process is implemented using a memory-efficient approach defined by the following parameters:

$$\text{Batch size} = 4 \tag{38}$$

$$\text{Input resolution} = 416 \times 416 \text{ pixels} \tag{39}$$

$$\text{Mixed precision training} = \text{Enabled} \tag{40}$$

$$\text{Worker threads} = 0 \tag{41}$$

$$\text{Epochs} = 10 \tag{42}$$

$$\text{Early stopping patience} = 25 \tag{43}$$

$$\text{Checkpoint frequency} = 5 \text{ epochs} \tag{44}$$

 doi:10.20944/preprints202507.1307.v1

For YOLOv9, the model sizes examined include tiny (t), small (s), medium (m), compact (c), and extended (e) configurations. For YOLOv8 and YOLOv10, the analysis encompasses nano (n), small (s), medium (m), large (l), and extra-large (x) variants. This range of model scales enables a comprehensive evaluation of the trade-off between computational efficiency and detection performance.

The dataset organization adheres to the standard YOLO format, with distinct directories for training and validation sets, each containing separate folders for images and corresponding label annotations. The label format follows the YOLO convention with normalized bounding box coordinates and class identifiers that map to the predefined categories of insulator conditions: intact insulators, flashover-damaged insulator shells, and broken insulator shells.

5.2. Implementation Details

The implementation leverages the Ultralytics YOLO framework with custom modifications to incorporate the ACR module. The training pipeline includes the following components:

1. **Data Configuration**: A custom YAML configuration generator that creates the required dataset specification with appropriate paths and class definitions.
2. **Memory Management**: Implementation of memory optimization techniques including:
   - Dynamic garbage collection
   - CUDA cache emptying between training runs
   - Automatic mixed precision (AMP) to reduce memory footprint
   - Minimal worker threads to reduce parallel processing overhead
3. **Adaptive Training**: The framework includes fallback mechanisms to handle potential memory limitations:
   - Automatic reduction of batch size and image resolution if out-of-memory errors occur
   - Optional subset training on reduced dataset samples for preliminary model validation
   - Device-agnostic implementation with CPU fallback capability
4. **Evaluation Metrics**: The training process tracks multiple performance indicators including:
   - Precision and recall for class-specific performance
   - mAP at IoU thresholds of 0.5 (mAP@0.5) and 0.5:0.95 (mAP@0.5:0.95)
   - F1-score for balanced evaluation of precision and recall

The model training is executed with consistent hyperparameters across all YOLO versions to ensure fair comparison. The base models are initialized with pre-trained weights, and then fine-tuned on the insulator dataset with the addition of the ACR module.

5.3. Experimental Framework

The experimental evaluation follows a structured protocol to assess the contribution of the proposed ACR enhancement:

1. **Baseline Evaluation**: Each YOLO model variant is first trained without the ACR module to establish baseline performance metrics.

2. **Enhanced Model Training**: The same model architectures are then augmented with the ACR module and trained using identical datasets and hyperparameters.

5.4. Baseline Analysis with YOLO Variants

To evaluate the baseline performance of state-of-the-art YOLO models, we conducted a comparative study across 25 variants spanning the YOLOv8, YOLOv9, YOLOv10, YOLOv11, and YOLOv12 families. Each model was trained under identical settings using the ACF detection pipeline, and their final detection performance metrics are summarized in Table 1. All models were trained for 5 epochs each, in order to keep the comparisons fair.

 doi:10.20944/preprints202507.1307.v1

Table 1. Final YOLO-ACF Metrics.

Model	Final mAP	Final Precision	Final Recall
yolo8l	0.673160	0.765651	0.820825
yolo8m	0.717706	0.820685	0.845983
yolo8n	0.634158	0.707193	0.805160
yolo8s	0.690742	0.854654	0.773468
yolo8x	0.681275	0.812279	0.807037
yolov9c	0.691740	0.807288	0.841653
yolov9e	0.671942	0.772708	0.779811
yolov9m	0.695113	0.824548	0.811013
yolov9s	0.696710	0.792208	0.826039
yolov9t	0.606963	0.700928	0.768216
yolov10l	0.664870	0.778065	0.781277
yolov10m	0.669379	0.812874	0.762396
yolov10n	0.556448	0.621034	0.685595
yolov10s	0.662223	0.799026	0.782245
yolov10x	0.666366	0.788095	0.758736
yolo11l	0.666793	0.774235	0.808157
yolo11m	0.682998	0.788911	0.843897
yolo11n	0.607671	0.701113	0.720074
yolo11s	0.686071	0.826978	0.808819
yolo11x	0.637785	0.796459	0.772851
yolo12l	0.649314	0.756868	0.778446
yolo12m	0.646290	0.731834	0.800345
yolo12n	0.610553	0.715652	0.756351
yolo12s	0.675353	0.776610	0.803838
yolo12x	0.614661	0.709647	0.782829

The highest mAP was achieved by `yolo8m` (0.718), followed closely by `yolov9s` and `yolov9m`, indicating that medium-sized models in the YOLOv8 and YOLOv9 families yield the best trade-off between capacity and generalization. Interestingly, smaller models such as `yolo8s` also perform competitively (0.691 mAP), demonstrating strong precision (0.855) despite slightly lower recall (0.773). This suggests that `yolo8s` is a conservative detector, generating fewer false positives, which is often desirable in safety-critical applications.

To further analyze the trade-offs between precision and recall across models, we plotted all configurations on a scatter diagram in Figure 4, with the mAP score represented as color intensity and the Pareto frontier overlaid.

Figure 4 illustrates that only a subset of models lie on the Pareto frontier, including `yolo8s`, `yolov9m`, `yolo8m`, and `yolov9c`. These models provide the most favorable trade-offs: no other configuration simultaneously improves both precision and recall compared to them. In contrast, models such as `yolov10n` and `yolo11n` exhibit significantly lower performance in both metrics and are Pareto-dominated.

 doi:10.20944/preprints202507.1307.v1

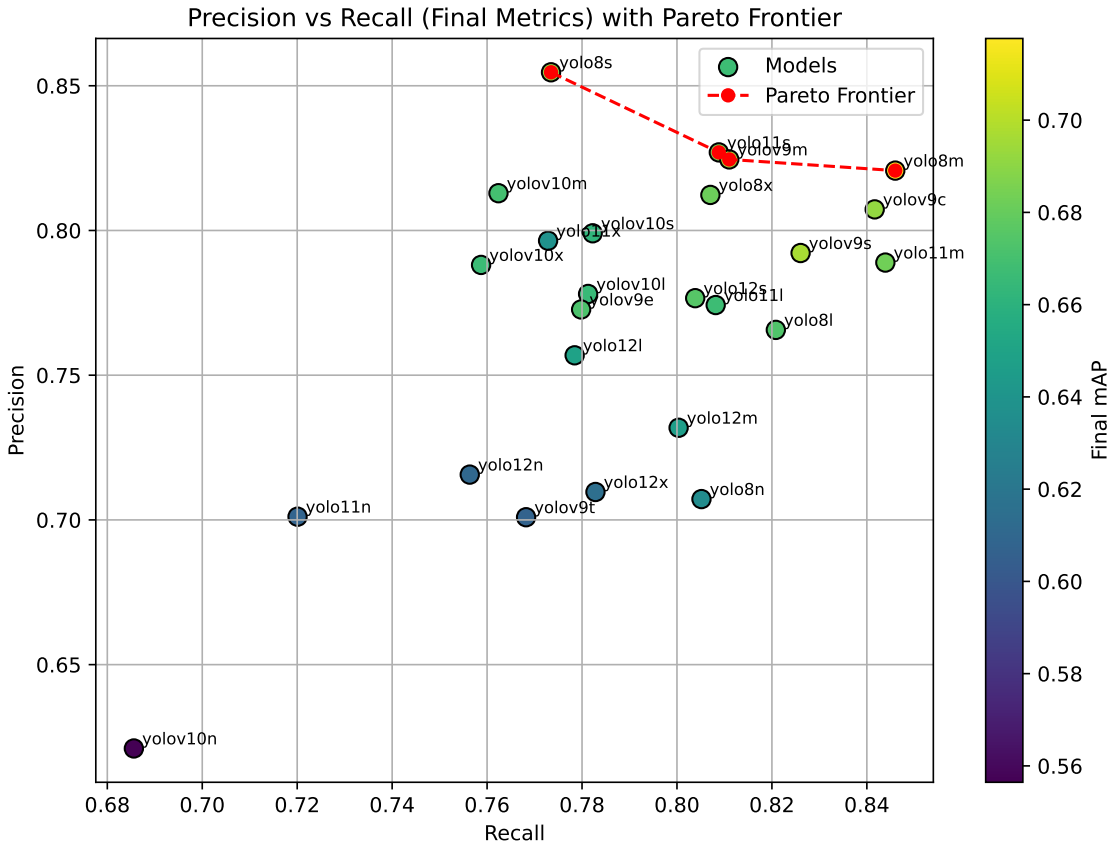

Figure 4. Precision vs. Recall across YOLO variants. Each point is colored by the final mAP. The red dashed line represents the Pareto frontier, models not dominated in both precision and recall.

5.5. Refinement Analysis and Cross-Generation Comparison

To assess the effectiveness of our proposed refinement methodology, we applied the optimized training pipeline to all 25 YOLO variants and compared their performance against the baseline results. Table 2 presents a comprehensive comparison of original versus refined mAP scores, including absolute improvements and percentage gains for each model configuration.

The refinement methodology demonstrates varying degrees of effectiveness across different YOLO generations. Most notably, **YOLOv10 exhibits the most substantial improvements**, with an average gain of approximately 12.6% across all model sizes. The standout performer is `yolov10n`, which achieved a remarkable 22.9% improvement (from 0.556 to 0.684 mAP), the largest single improvement in the entire evaluation. This dramatic enhancement suggests that YOLOv10's architecture is particularly well-suited to benefit from our proposed refinement approach, especially for resource-constrained nano variants that initially exhibited lower baseline performance.

YOLOv12 demonstrates consistent substantial gains with an average improvement of approximately 8.6%, indicating that the newest generation also responds favorably to the refinement methodology. Notably, `yolo12x` achieved an 11.96% improvement, while the smaller variants (`yolo12n` and `yolo12m`) showed gains of 9.90% and 9.35%, respectively.

In contrast, **YOLOv8 and YOLOv9 exhibit more modest improvements**, with average gains of 4.0% and 5.6%, respectively. However, within these families, smaller models tend to benefit more significantly than their larger counterparts. For instance, `yolo8n` improved by 8.36%, while `yolo8m`, the baseline champion-gained only 0.84%. This pattern suggests that well-optimized larger models may already be operating closer to their performance ceiling, whereas smaller models have greater potential for improvement through refined training strategies.

YOLOv11 shows mixed results, with an average improvement of 5.4%. Interestingly, `yolo11x` achieved a substantial 9.99% gain, while `yolo11s` showed minimal improvement (0.97%). This vari-

 doi:10.20944/preprints202507.1307.v1

ability within the YOLOv11 family indicates that the effectiveness of our refinement approach may depend on specific architectural characteristics and initial optimization states.

Table 2. Comparison of Original vs. Refined YOLO-ACF Performance

Model	Original mAP	Refined mAP	δ mAP	% Gain
yolo8l	0.673160	0.703000	0.029840	4.43
yolo8m	0.717706	0.723700	0.005994	0.84
yolo8n	0.634158	0.687200	0.053042	8.36
yolo8s	0.690742	0.714600	0.023858	3.45
yolo8x	0.681275	0.702600	0.021325	3.13
yolov9c	0.691740	0.724500	0.032760	4.74
yolov9e	0.671942	0.702400	0.030458	4.53
yolov9m	0.695113	0.724400	0.029287	4.21
yolov9s	0.696710	0.722500	0.025790	3.70
yolov9t	0.606963	0.673300	0.066337	10.93
yolov10l	0.664870	0.745300	0.080430	12.10
yolov10m	0.669379	0.723600	0.054221	8.10
yolov10n	0.556448	0.683800	0.127352	22.89
yolov10s	0.662223	0.730800	0.068577	10.36
yolov10x	0.666366	0.732200	0.065834	9.88
yolo11l	0.666793	0.709700	0.042907	6.43
yolo11m	0.682998	0.700700	0.017702	2.59
yolo11n	0.607671	0.651200	0.043529	7.16
yolo11s	0.686071	0.692700	0.006629	0.97
yolo11x	0.637785	0.701500	0.063715	9.99
yolo12l	0.649314	0.698100	0.048786	7.51
yolo12m	0.646290	0.706700	0.060410	9.35
yolo12n	0.610553	0.671000	0.060447	9.90
yolo12s	0.675353	0.705400	0.030047	4.45
yolo12x	0.614661	0.688200	0.073539	11.96

The results reveal an important trend: **models with initially lower baseline performance tend to benefit more from refinement**. This is particularly evident in the nano variants across all generations, where substantial improvements were consistently observed. Conversely, models that already achieved high baseline performance, such as `yolo8m` and `yolo11s`, showed diminishing returns from further optimization, suggesting they may have already converged to near-optimal solutions under the original training regime.

6. Conclusions

This study evaluates YOLO-based insulator fault detection systems enhanced with ACR across 25 model variants spanning YOLOv8 through YOLOv12. The baseline analysis established performance characteristics for each architecture, while the refinement methodology demonstrated varying degrees of improvement across different YOLO generations.

The baseline evaluation identified `yolo8m` as achieving the highest initial performance with 0.718 mAP. Pareto frontier analysis revealed four configurations (`yolo8s`, `yolov9m`, `yolo8m`, and `yolov9c`) that provide optimal precision-recall trade-offs, with no other models simultaneously improving both metrics.

The ACR refinement methodology produced the largest improvements in YOLOv10, with an average gain of 12.6% across all model sizes. The `yolov10n` variant achieved the maximum individual improvement of 22.9% (from 0.556 to 0.684 mAP). YOLOv12 models showed consistent gains averaging 8.6%, while YOLOv8 and YOLOv9 exhibited more modest improvements of 4.0% and 5.6%, respectively.

 doi:10.20944/preprints202507.1307.v1

The results reveal an inverse relationship between baseline performance and refinement potential: models with lower initial performance consistently achieved larger improvements through the ACR methodology. Nano variants across all generations demonstrated the highest percentage gains, while well-optimized models such as `yolo8m` and `yolo11s` showed minimal improvement, suggesting convergence to near-optimal solutions under the original training regime.

The proposed ACR-enhanced YOLO framework addresses key limitations in automated insulator inspection by incorporating contextual information to reduce false positives, improving detection of partial defects through adjacent area analysis, and enhancing classification accuracy by considering relative anomaly positions within insulator strings. The selective refinement process maintains computational efficiency by focusing processing resources on regions of interest, enabling practical deployment in field inspection systems.

Future work should investigate the generalization of ACR across different power grid components and explore adaptive context scale selection mechanisms based on environmental conditions and defect characteristics. Additionally, real-time deployment studies are needed to validate the practical performance of ACR-enhanced models in operational inspection scenarios.

Author Contributions: Writing—original draft: Aquino, L.S., and Agottani, L.F.R.; writing—review and editing, software, methodology, validation: Seman, L.O.; writing—review and editing, supervision, Mariani, V.C.; Coelho, L.S.; and González, G. V. All authors have read and agreed to the published version of the manuscript.

Funding: This work is part of the project Self-adaptive platform based on intelligent agents for the optimization and management of operational processes in logistic warehouses (PLAUTON) PID2023-151701OB-C21, funded by MCIN/AEI/10.13039/501100011033/FEDER, EU.

Informed Consent Statement: Not applicable.

Data Availability Statement: Available upon request to the authors.

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

References

1. Ahmed, M.; Mohanta, J.; Sanyal, A. Inspection and identification of transmission line insulator breakdown based on deep learning using aerial images. *Electric Power Systems Research* **2022**, *211*, 108199. https://doi.org/10.1016/j.epsr.2022.108199.
2. Medeiros, A.; Sartori, A.; Stefenon, S.F.; Meyer, L.H.; Nied, A. Comparison of artificial intelligence techniques to failure prediction in contaminated insulators based on leakage current. *Journal of Intelligent & Fuzzy Systems* **2022**, *42*, 3285–3298. https://doi.org/10.3233/JIFS-211126.
3. Arias Velásquez, R.M. Insulation failure caused by special pollution around industrial environments. *Engineering Failure Analysis* **2019**, *102*, 123–135. https://doi.org/10.1016/j.engfailanal.2019.04.034.
4. Saberironaghi, A.; Ren, J.; El-Gindy, M. Defect Detection Methods for Industrial Products Using Deep Learning Techniques: A Review. *Algorithms* **2023**, *16*. https://doi.org/10.3390/a16020095.
5. Seman, L.O.; Stefenon, S.F.; Mariani, V.C.; Coelho, L.S. Ensemble learning methods using the Hodrick–Prescott filter for fault forecasting in insulators of the electrical power grids. *International Journal of Electrical Power & Energy Systems* **2023**, *152*, 109269. https://doi.org/10.1016/j.ijepes.2023.109269.
6. Stefenon, S.F.; Oliveira, J.R.; Coelho, A.S.; Meyer, L.H. Diagnostic of insulators of conventional grid through LabVIEW analysis of FFT signal generated from ultrasound detector. *IEEE Latin America Transactions* **2017**, *15*, 884–889. https://doi.org/10.1109/TLA.2017.7910202.
7. Aljalaud, F.; Kurdi, H.; Youcef-Toumi, K. Autonomous Multi-UAV Path Planning in Pipe Inspection Missions Based on Booby Behavior. *Mathematics* **2023**, *11*. https://doi.org/10.3390/math11092092.
8. Gong, Y.; Zhou, W.; Wang, K.; Wang, J.; Wang, R.; Deng, H.; Liu, G. Defect detection of small cotter pins in electric power transmission system from UAV images using deep learning techniques. *Electrical Engineering* **2023**, *105*, 1251 – 1266. https://doi.org/10.1007/s00202-022-01729-8.
9. Corso, M.P.; Stefenon, S.F.; Singh, G.; Matsuo, M.V.; Perez, F.L.; Leithardt, V.R.Q. Evaluation of visible contamination on power grid insulators using convolutional neural networks. *Electrical Engineering* **2023**, *105*, 3881–3894. https://doi.org/10.1007/s00202-023-01915-2.

 doi:10.20944/preprints202507.1307.v1

10. Yi, J.; Mao, J.; Zhang, H.; Chen, Y.; Liu, T.; Zeng, K.; Xie, H.; Wang, Y. Balancing Accuracy and Efficiency With a Multiscale Uncertainty-Aware Knowledge-Based Network for Transmission Line Inspection. *IEEE Transactions on Industrial Informatics* **2025**, *21*, 2829–2838. https://doi.org/10.1109/TII.2024.3507936.
11. Labrador Rivas, A.E.; Abrão, T. Faults in smart grid systems: Monitoring, detection and classification. *Electric Power Systems Research* **2020**, *189*, 106602. https://doi.org/10.1016/j.epsr.2020.106602.
12. Preduș, M.F.; Popescu, C.; Răduca, E.; Hațiegan, C. Study of the Accelerated Degradation of the Insulation of Power Cables under the Action of the Acid Environment. *Energies* **2022**, *15*. https://doi.org/10.3390/en15103550.
13. You, X.; Zhao, X. A insulator defect detection network based on improved YOLOv7 for UAV aerial images. *Measurement* **2025**, *253*, 117410. https://doi.org/https://doi.org/10.1016/j.measurement.2025.117410.
14. Stefenon, S.F.; Singh, G.; Souza, B.J.; Freire, R.Z.; Yow, K.C. Optimized hybrid YOLOu-Quasi-ProtoPNet for insulators classification. *IET Generation, Transmission & Distribution* **2023**, *17*, 3501–3511. https://doi.org/10.1049/gtd2.12886.
15. Stefenon, S.F.; Yow, K.C.; Nied, A.; Meyer, L.H. Classification of distribution power grid structures using inception v3 deep neural network. *Electrical Engineering* **2022**, *104*, 4557–4569. https://doi.org/10.1007/s00202-022-01641-1.
16. Freitas-Gutierres, L.F.; Maresch, K.; Morais, A.M.; Nunes, M.V.; Correa, C.H.; Martins, E.F.; Fontoura, H.C.; Schmidt, M.V.; Soares, S.N.; Cardoso, G.; et al. Framework for decision-making in preventive maintenance: Electric field analysis and partial discharge diagnosis of high-voltage insulators. *Electric Power Systems Research* **2024**, *233*, 110447. https://doi.org/10.1016/j.epsr.2024.110447.
17. Mahapatra, U.; Rahman, M.; Islam, M.R.; Hossain, M.A.; Sheikh, M.R.I.; Hossain, M. Adversarial training-based robust model for transmission line's insulator defect classification against cyber-attacks. *Electric Power Systems Research* **2025**, *245*, 111585. https://doi.org/10.1016/j.epsr.2025.111585.
18. Stefenon, S.F.; Seman, L.O.; Pavan, B.A.; Ovejero, R.G.; Leithardt, V.R.Q. Optimal design of electrical power distribution grid spacers using finite element method. *IET Generation, Transmission & Distribution* **2022**, *16*, 1865–1876. https://doi.org/10.1049/gtd2.12425.
19. Stefenon, S.F.; Americo, J.P.; Meyer, L.H.; Grebogi, R.B.; Nied, A. Analysis of the electric field in porcelain pin-type insulators via finite elements software. *IEEE Latin America Transactions* **2018**, *16*, 2505–2512. https://doi.org/10.1109/TLA.2018.8795129.
20. Liao, W.; Zhu, R.; Ge, L.; Cao, D.; Yang, Z. Mitigating Class Imbalance Issues in Electricity Theft Detection via a Sample-Weighted Loss. *IEEE Transactions on Industrial Informatics* **2025**, *21*, 1754 – 1763. https://doi.org/10.1109/TII.2024.3485813.
21. Stefenon, S.F.; Seman, L.O.; Sopelsa Neto, N.F.; Meyer, L.H.; Mariani, V.C.; Coelho, L.d.S. Group method of data handling using Christiano-Fitzgerald random walk filter for insulator fault prediction. *Sensors* **2023**, *23*, 6118. https://doi.org/10.3390/s23136118.
22. Ren, Q.; Kang, W.; Yang, X.; Wang, Q.; Huang, Q. Intelligent recognition and sustainable security protection strategies for abnormal behavior of power grid operation data based on multidimensional digital portrait and deep neural networks. *Discover Artificial Intelligence* **2025**, *5*. https://doi.org/10.1007/s44163-025-00239-3.
23. Jiang, H.; Qiu, X.; Chen, J.; Liu, X.; Miao, X.; Zhuang, S. Insulator Fault Detection in Aerial Images Based on Ensemble Learning With Multi-Level Perception. *IEEE Access* **2019**, *7*, 61797–61810. https://doi.org/10.1109/ACCESS.2019.2915985.
24. Li, Y.; Zhu, C.; Zhang, Q.; Zhang, J.; Wang, G. IF-YOLO: An Efficient and Accurate Detection Algorithm for Insulator Faults in Transmission Lines. *IEEE Access* **2024**, *12*, 167388–167403. https://doi.org/10.1109/ACCESS.2024.3496514.
25. Souza, B.J.; Stefenon, S.F.; Singh, G.; Freire, R.Z. Hybrid-YOLO for classification of insulators defects in transmission lines based on UAV. *International Journal of Electrical Power & Energy Systems* **2023**, *148*, 108982. https://doi.org/10.1016/j.ijepes.2023.108982.
26. Zhao, W.; Xu, M.; Cheng, X.; Zhao, Z. An Insulator in Transmission Lines Recognition and Fault Detection Model Based on Improved Faster RCNN. *IEEE Transactions on Instrumentation and Measurement* **2021**, *70*, 1–8. https://doi.org/10.1109/TIM.2021.3112227.
27. Song, C.; Xu, W.; Wang, Z.; Yu, S.; Zeng, P.; Ju, Z. Analysis on the Impact of Data Augmentation on Target Recognition for UAV-Based Transmission Line Inspection. *Complexity* **2020**, *2020*. https://doi.org/10.1155/2020/3107450.
28. Han, G.; Li, T.; Li, Q.; Zhao, F.; Zhang, M.; Wang, R.; Yuan, Q.; Liu, K.; Qin, L. Improved Algorithm for Insulator and Its Defect Detection Based on YOLOX. *Sensors* **2022**, *22*. https://doi.org/10.3390/s22166186.

 doi:10.20944/preprints202507.1307.v1

29. Wang, S.; Liu, Y.; Qing, Y.; Wang, C.; Lan, T.; Yao, R. Detection of Insulator Defects With Improved ResNeSt and Region Proposal Network. *IEEE Access* **2020**, *8*, 184841–184850. https://doi.org/10.1109/ACCESS.2020.3029857.
30. Shuang, F.; Wei, S.; Li, Y.; Gu, X.; Lu, Z. Detail R-CNN: Insulator Detection Based on Detail Feature Enhancement and Metric Learning. *IEEE Transactions on Instrumentation and Measurement* **2023**, *72*, 1–14. https://doi.org/10.1109/TIM.2023.3305667.
31. Zhou, M.; Wang, J.; Li, B. ARG-Mask RCNN: An Infrared Insulator Fault-Detection Network Based on Improved Mask RCNN. *Sensors* **2022**, *22*. https://doi.org/10.3390/s22134720.
32. Liang, H.; Zuo, C.; Wei, W. Detection and Evaluation Method of Transmission Line Defects Based on Deep Learning. *IEEE Access* **2020**, *8*, 38448 – 38458. https://doi.org/10.1109/ACCESS.2020.2974798.
33. Chen, H.; He, Z.; Shi, B.; Zhong, T. Research on Recognition Method of Electrical Components Based on YOLO V3. *IEEE Access* **2019**, *7*, 157818 – 157829. https://doi.org/10.1109/ACCESS.2019.2950053.
34. Han, J.; Yang, Z.; Xu, H.; Hu, G.; Zhang, C.; Li, H.; Lai, S.; Zeng, H. Search like an eagle: A cascaded model for insulator missing faults detection in aerial images. *Energies* **2020**, *13*. https://doi.org/10.3390/en13030713.
35. Costa, A.; Cortes, C.O. A Convolutional Neural Network for Detecting Faults in Power Distribution Networks along a Railway: A Case Study Using YOLO. *Applied Artificial Intelligence* **2021**, *35*, 2067 – 2086. https://doi.org/10.1080/08839514.2021.1998974.
36. Qiu, Z.; Zhu, X.; Liao, C.; Shi, D.; Qu, W. Detection of Transmission Line Insulator Defects Based on an Improved Lightweight YOLOv4 Model. *Applied Sciences (Switzerland)* **2022**, *12*. https://doi.org/10.3390/app12031207.
37. Liquan, Z.; Mengjun, Z.; Ying, C.; Yanfei, J. Fast Detection of Defective Insulator Based on Improved YOLOv5s. *Computational Intelligence and Neuroscience* **2022**, *2022*. https://doi.org/10.1155/2022/8955292.
38. Zhang, T.; Zhang, Y.; Xin, M.; Liao, J.; Xie, Q. A Light-Weight Network for Small Insulator and Defect Detection Using UAV Imaging Based on Improved YOLOv5. *Sensors* **2023**, *23*. https://doi.org/10.3390/s23115249.
39. Zheng, J.; Wu, H.; Zhang, H.; Wang, Z.; Xu, W. Insulator-Defect Detection Algorithm Based on Improved YOLOv7. *Sensors* **2022**, *22*. https://doi.org/10.3390/s22228801.
40. Chen, B.; Zhang, W.; Wu, W.; Li, Y.; Chen, Z.; Li, C. ID-YOLOv7: an efficient method for insulator defect detection in power distribution network. *Frontiers in Neurorobotics* **2023**, *17*. https://doi.org/10.3389/fnbot.2023.1331427.
41. Zhang, J.; Wei, X.; Zhang, L.; Yu, L.; Chen, Y.; Tu, M. YOLO v7-ECA-PConv-NWD Detects Defective Insulators on Transmission Lines. *Electronics (Switzerland)* **2023**, *12*. https://doi.org/10.3390/electronics12183969.
42. Gong, C.; Jiang, W.; Zou, D.; Weng, W.; Li, H. An Insulator Fault Diagnosis Method Based on Multi-Mechanism Optimization YOLOv8. *Applied Sciences* **2024**, *14*. https://doi.org/10.3390/app14198770.
43. Wang, G.; Chen, Y.; An, P.; Hong, H.; Hu, J.; Huang, T. UAV-YOLOv8: A Small-Object-Detection Model Based on Improved YOLOv8 for UAV Aerial Photography Scenarios. *Sensors* **2023**, *23*. https://doi.org/10.3390/s23167190.
44. Tao, Z.; He, Y.; Lin, S.; Yi, T.; Li, M. SnakeNet: An adaptive network for small object and complex background for insulator surface defect detection. *Computers and Electrical Engineering* **2024**, *117*, 109259. https://doi.org/10.1016/j.compeleceng.2024.109259.
45. He, M.; Qin, L.; Deng, X.; Liu, K. MFI-YOLO: Multi-Fault Insulator Detection Based on an Improved YOLOv8. *IEEE Transactions on Power Delivery* **2024**, *39*, 168–179. https://doi.org/10.1109/TPWRD.2023.3328178.
46. Wang, Y.; Song, X.; Feng, L.; Zhai, Y.; Zhao, Z.; Zhang, S.; Wang, Q. MCI-GLA Plug-In Suitable for YOLO Series Models for Transmission Line Insulator Defect Detection. *IEEE Transactions on Instrumentation and Measurement* **2024**, *73*, 1–12. https://doi.org/10.1109/TIM.2024.3385817.
47. Li, J.; Zhou, H.; Lv, G.; Chen, J. A2MADA-YOLO: Attention Alignment Multiscale Adversarial Domain Adaptation YOLO for Insulator Defect Detection in Generalized Foggy Scenario. *IEEE Transactions on Instrumentation and Measurement* **2025**, *74*, 1–19. https://doi.org/10.1109/TIM.2025.3541814.
48. Sadykova, D.; Pernebayeva, D.; Bagheri, M.; James, A. IN-YOLO: Real-Time Detection of Outdoor High Voltage Insulators Using UAV Imaging. *IEEE Transactions on Power Delivery* **2020**, *35*, 1599–1601. https://doi.org/10.1109/TPWRD.2019.2944741.
49. Cao, Z.; Chen, K.; Chen, J.; Chen, Z.; Zhang, M. CACS-YOLO: A Lightweight Model for Insulator Defect Detection Based on Improved YOLOv8m. *IEEE Transactions on Instrumentation and Measurement* **2024**, *73*, 1–10. https://doi.org/10.1109/TIM.2024.3453332.

 doi:10.20944/preprints202507.1307.v1

50. Zhang, Q.; Zhang, J.; Li, Y.; Zhu, C.; Wang, G. IL-YOLO: An Efficient Detection Algorithm for Insulator Defects in Complex Backgrounds of Transmission Lines. *IEEE Access* **2024**, *12*, 14532–14546. https://doi.org/10.1109/ACCESS.2024.3358205.
51. Zhang, Q.; Zhang, J.; Li, Y.; Zhu, C.; Wang, G. ID-YOLO: A Multimodule Optimized Algorithm for Insulator Defect Detection in Power Transmission Lines. *IEEE Transactions on Instrumentation and Measurement* **2025**, *74*, 1–11. https://doi.org/10.1109/TIM.2025.3527530.
52. Ding, L.; Rao, Z.Q.; Ding, B.; Li, S.J. Research on Defect Detection Method of Railway Transmission Line Insulators Based on GC-YOLO. *IEEE Access* **2023**, *11*, 102635–102642. https://doi.org/10.1109/ACCESS.2023.3316266.
53. Li, D.; Lu, Y.; Gao, Q.; Li, X.; Yu, X.; Song, Y. LiteYOLO-ID: A Lightweight Object Detection Network for Insulator Defect Detection. *IEEE Transactions on Instrumentation and Measurement* **2024**, *73*, 1–12. https://doi.org/10.1109/TIM.2024.3418082.
54. Li, Y.; Feng, D.; Zhang, Q.; Li, S. HRD-YOLOX Based Insulator Identification and Defect Detection Method for Transmission Lines. *IEEE Access* **2024**, *12*, 22649–22661. https://doi.org/10.1109/ACCESS.2024.3363430.
55. Hu, M.; Liu, J.; Liu, J. DRR-YOLO: A Study of Small Target Multi-Modal Defect Detection for Multiple Types of Insulators Based on Large Convolution Kernel. *IEEE Access* **2025**, *13*, 26331–26344. https://doi.org/10.1109/ACCESS.2025.3539831.
56. Liu, B.; Jiang, W. LA-YOLO: Bidirectional Adaptive Feature Fusion Approach for Small Object Detection of Insulator Self-Explosion Defects. *IEEE Transactions on Power Delivery* **2024**, *39*, 3387–3397. https://doi.org/10.1109/TPWRD.2024.3467915.
57. Shen, P.; Mei, K.; Cao, H.; Zhao, Y.; Zhang, G. LDDFSF-YOLO11: A Lightweight Insulator Defect Detection Method Focusing on Small-Sized Features. *IEEE Access* **2025**, *13*, 90273–90292. https://doi.org/10.1109/ACCESS.2025.3569970.
58. Zhang, L.; Li, B.; Cui, Y.; et al. Research on improved YOLOv8 algorithm for insulator defect detection. *Journal of Real-Time Image Processing* **2024**, *21*. https://doi.org/10.1007/s11554-023-01401-9.
59. Zhao, J.; Miao, S.; Kang, R.; Cao, L.; Zhang, L.; Ren, Y. Insulator Defect Detection Algorithm Based on Improved YOLOv11n. *Sensors* **2025**, *25*. https://doi.org/10.3390/s25051327.
60. Stefenon, S.F.; Seman, L.O.; Singh, G.; Yow, K.C. Enhanced insulator fault detection using optimized ensemble of deep learning models based on weighted boxes fusion. *International Journal of Electrical Power & Energy Systems* **2025**, *168*, 110682. https://doi.org/10.1016/j.ijepes.2025.110682.
61. Xu, R.; Zheng, Y.; Wang, X.; Li, D. Person re-identification based on improved attention mechanism and global pooling method. *Journal of Visual Communication and Image Representation* **2023**, *94*, 103849. https://doi.org/10.1016/j.jvcir.2023.103849.
62. Singh, G.; Stefenon, S.F.; Yow, K.C. Interpretable visual transmission lines inspections using pseudo-prototypical part network. *Machine Vision and Applications* **2023**, *34*, 41. https://doi.org/10.1007/s00138-023-01390-6.
63. Tan, X.; Hou, S.; Yang, F.; Li, Z. Transmission Lines Insulator State Detection Method Based on Deep Learning. *Applied Sciences* **2025**, *15*. https://doi.org/10.3390/app15020526.
64. Stefenon, S.F.; Cristoforetti, M.; Cimatti, A. Towards Automatic Digitalization of Railway Engineering Schematics. In Proceedings of the AIxIA 2023 – Advances in Artificial Intelligence, Rome, Italy, 2023; Vol. 22, pp. 453–466. https://doi.org/10.1007/978-3-031-47546-7_31.
65. Lu, G.; Li, B.; Chen, Y.; Qu, S.; Cheng, T.; Zhou, J. Precision in Aerial Surveillance: Integrating YOLOv8 With PConv and CoT for Accurate Insulator Defect Detection. *IEEE Access* **2025**, *13*, 49062–49075. https://doi.org/10.1109/ACCESS.2025.3551289.
66. Panigrahy, S.; Karmakar, S. Real-Time Condition Monitoring of Transmission Line Insulators Using the YOLO Object Detection Model With a UAV. *IEEE Transactions on Instrumentation and Measurement* **2024**, *73*, 1–9. https://doi.org/10.1109/TIM.2024.3381693.
67. Stefenon, S.F.; Seman, L.O.; Klaar, A.C.R.; Ovejero, R.G.; Leithardt, V.R.Q. Hypertuned-YOLO for interpretable distribution power grid fault location based on EigenCAM. *Ain Shams Engineering Journal* **2024**, *15*, 102722. https://doi.org/10.1016/j.asej.2024.102722.
68. Stefenon, S.F.; Cristoforetti, M.; Cimatti, A. Automatic digitalization of railway interlocking systems engineering drawings based on hybrid machine learning methods. *Expert Systems with Applications* **2025**, *281*, 127532. https://doi.org/10.1016/j.eswa.2025.127532.

 doi:10.20944/preprints202507.1307.v1

69. Lewis, D.; Kulkarni, P. Insulator Defect Detection. https://ieee-dataport.org/competitions/insulator-defect-detection, 2021. [Accessed 8 May 2025].
70. Stefenon, S.F. Inspection of Electrical Power Distribution Grid. https://github.com/SFStefenon/InspectionDataSet, 2023. [Accessed 8 May 2025].