Detection Of Power Line Insulator Defects Using YOLOv10-N

Kgampu Shawn Papi[1] and Terence van Zyl[1]

1 Institute for Intelligent Systems, University of Johannesburg, Johannesburg, South Africa
papiks@eskom.co.za, tvanzyl@uj.ca.za

Abstract. Visual inspection remains a common approach for assessing composite insulators, with unmanned aerial vehicles (UAVs) increasingly preferred due to their efficiency and reduced error rates. Recent developments have integrated artificial intelligence (AI) algorithms directly into UAV hardware to enable faster processing; however, such systems require optimized models owing to limited onboard computing resources. The recently introduced YOLOv10-N model, which offers greater efficiency compared to its predecessors, demonstrates potential for detecting insulator defects on resource-constrained UAV platforms. This study evaluates the effectiveness of YOLOv10-N for this application.

Keywords: YOLO, Insulator, Visual inspection, Image, Embedded systems.

1 Introduction

Visual inspection remains one of the most widely used methods for assessing the condition of composite insulators. This technique is generally effective, as many types of surface damage can be readily identified during climbing inspections or, in some cases, from the ground, a helicopter, or an unmanned aerial vehicle (UAV). UAV technology is rapidly becoming the preferred method for aerial insulator inspection because traditional alternatives, such as helicopter-based surveys, are time-consuming, costly, and susceptible to human error [1].

Recent developments indicate a shift toward integrating artificial intelligence (AI) applications directly into UAV hardware or single-board computers mounted on the UAV. This approach offers lower latency compared to UAV systems that stream imagery to cloud-based AI applications for processing [2]. UAVs with onboard AI capabilities are receiving growing research attention in insulator inspection applications [3, 4]. However, single-board computers (e.g., Raspberry Pi) and UAV hardware generally have limited processing power, restricting them to AI models optimized for devices with constrained computational resources.

Panigrahy et al. [4] deployed a YOLOv8-N model on a Raspberry Pi 4 mounted on a UAV and demonstrated the model's ability to detect insulators under varying climatic conditions. Notably, the recently introduced YOLOv10-N model requires 28% fewer parameters than YOLOv8-N for object detection [5], thereby reducing the

computational requirements for UAV-mounted embedded systems. In this paper, the performance of YOLOv10-N is evaluated for the task of insulator defect detection.

1.1 Related Work

Extensive research has examined the application of deep learning algorithms for insulator defect detection. As noted by Liu et al. [6], most studies focus on improving model speed and accuracy, with comparatively limited consideration of the computational constraints of the target deployment hardware. More recently, several studies have explored models designed for execution on computers mounted on, or embedded within, unmanned aerial vehicles (UAVs). Deploying defect detection directly on the UAV offers several advantages, including reduced latency, improved operational efficiency, enhanced reliability, and greater data privacy [2].

It has been demonstrated that a trained YOLOv8-N model can operate on a single-board computer mounted on a UAV and successfully perform real-time insulator defect detection under varying climatic conditions. The study employed a self-curated dataset comprising 6020 porcelain insulator images, augmented using 32 data augmentation techniques to improve detection accuracy [4]. Bellou et al. [3] evaluated YOLOv8-S, a low-parameter model, on a self-curated dataset, and their results confirmed that lightweight architectures can achieve competitive performance for insulator defect detection. Lu et al. [6] observed that "most two-stage detection models fail to meet the real-time detection requirements in practical engineering and often occupy excessive memory during deployment. In contrast, one-stage detection models can support real-time detection, but their recognition accuracy still requires improvement." To address these limitations, they proposed IDD-YOLO (Insulator Defect Detection-YOLO), a lightweight object detection model based on YOLOv8. The IDD-YOLO backbone employs GhostNet for feature extraction, enhanced by the Lightweight Channel–Spatial Attention (LCSA) mechanism to improve feature extraction capability. The neck incorporates GSConv and C3Ghost convolution modules and uses PANet for feature transformation, thereby reducing unnecessary parameters and network complexity. To maximize detection speed and precision, IDD-YOLO retains the original YOLO detection head while adopting the EIOU loss function and Mish activation function.

The literature indicates a prevailing focus on one-stage object detection models, particularly YOLO variants, for UAV-based embedded deployments. Notably, while several studies published in 2024 continue to center on YOLOv8, YOLOv10—released the same year—offers further improvements in accuracy and inference speed with reduced parameter requirements.

1.2 The YOLOv10 Model

Ultralytics released YOLOv10 in May 2024, introducing several enhancements over YOLOv9. A key improvement is the removal of the dependency on non-maximum suppression (NMS). Previous YOLO models employed a one-to-many assignment strategy during training, producing multiple predicted bounding boxes for each inference and relying on NMS to eliminate redundant predictions, which increased inference latency. YOLOv10 also incorporates accuracy improvements, including the use of Large-

Kernel Convolution for enhanced feature extraction and Partial Self-Attention (PSA) to improve global representation learning with minimal computational overhead [5].

YOLOv10 is available in multiple model variants; in this work, only YOLOv10-N is evaluated, as it has the lowest parameter count. As shown in Table 1, YOLOv10-N requires 28% fewer parameters than YOLOv8-N while achieving superior speed and accuracy on the COCO dataset.

Table 1. Comparison of YOLOv8-N and YOLOv10-N.

Model	Parameters (Million)	AP^{val}(%)	Latency(ms)
YOLOv8-N	3.2	37.3	6.16
YOLOV10-N	2.3	38.5 / 39.5	1.84

2 Methodology

This section presents an overview of the tasks conducted in evaluating the YOLOv10-N model for insulator defect detection. The methodology adopted in this study is summarized in the flowchart shown in Fig. 1.

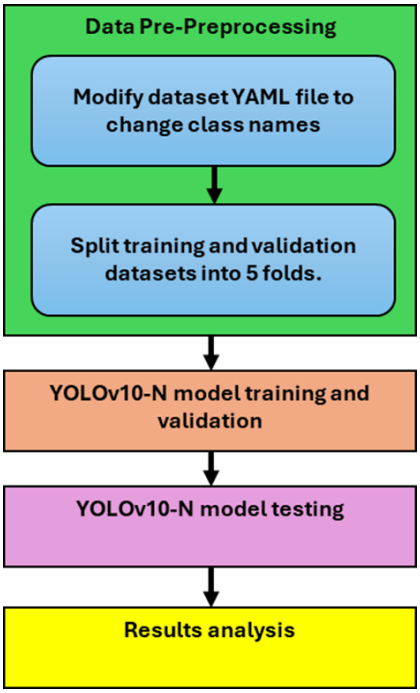

Fig. 1. Summary of tasks.

2.1 Dataset

The publicly available Insulator Defect Detection dataset, preprocessed by Fahim and Hasan, is utilized in this study [7, 8]. A sample of the dataset is shown in Fig. 2.

Fig. 2. Sample images from dataset.

The dataset comprises 1600 images, divided into a training set of 1280 images, a validation set of 235 images, and a test set of 85 images. To mitigate overfitting, the images include diverse backgrounds and various insulator disc arrangements. All images were resized to a standard resolution of 600 × 600 pixels for efficient training and are presented in different orientations. Annotations were performed in the YOLO format for four classes: insulator shell (I), broken insulator disc (B), flashover-damaged insulator disc (F), and good insulator disc (G) [9].

2.2 Experiment Setup

To ensure better generalization during inference on previously unseen data, the YOLOv10-N model was trained using K-fold cross-validation. The combined training and validation datasets were partitioned into five folds using Scikit-Learn's *KFold* class, with parameters set as follows: K=5K = 5K=5, *shuffle* = True, and *random_state* = 20 [10]. The model training was implemented in Python 3 on the Google Colab platform, employing the Ultralytics library and YOLOv10-N pretrained weights. Certain YOLOv10-N hyperparameters were set according to the values in Table 2, while the remaining parameters retained their default settings.

Hyperparameter	Value	Remark
Epochs	100	Default
Batch	-1	60% of the GPU RAM.
Image size	608	Must be multiple of 32,
Optimizer	AdamW	

Table 2. Training parameter settings.

A Google Colab runtime equipped with an NVIDIA Tesla T4 GPU was used for both training and validation of the YOLOv10-N model.

2.3 Evaluation Metrics

The performance of the YOLOv10-N model will be evaluated on the test set using three metrics, namely mean average precision (mAP), F1-confidence and the confusion matrix.

3 Results And Discussion

A YOLOv10-N model was trained and validated using the K-fold cross-validation technique and subsequently evaluated on the test dataset. This section presents and discusses the corresponding results.

A summary of the mAP50 scores for the YOLOv10-N model is provided in Table 3. The Precision–Recall curve is shown in Fig. 3, indicating that the model maintains a precision value close to 1.0 for most recall values. Overall, the model performs well; however, class B exhibits a comparatively low mAP50 score of 0.872.

Class	mAP50
All	0.934
B (Broken insulator disc)	0.872
F (Flash over damaged insulator disc)	0.945
G (Good insulator disc)	0.944
I (Insulator shell)	0.974

Table 3. Summary of evaluation results, mAP.

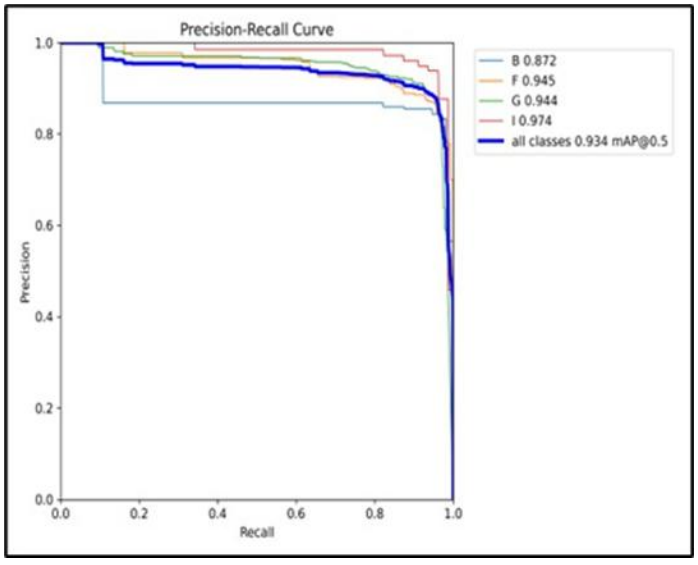

Fig. 3. Precision-Recall curve.

The F1–Confidence curve is presented in Fig. 4. The model achieves strong performance across all classes, with an overall F1 score of 0.92.

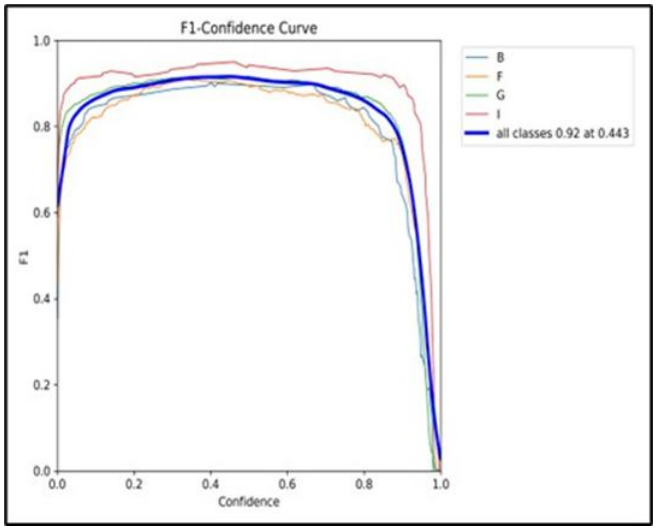

Fig. 4. F1-Confidence curve.

The confusion matrix in Fig. 5 shows balanced accuracy across most classes. Nonetheless, the model frequently misclassifies the background as class G. This misclassification may be attributed to the presence of G-class bounding boxes overlapping in images containing insulator discs with minimal separation. An example inference is provided in Fig. 6. This issue could be mitigated by adding more training images of this nature to ensure adequate feature learning.

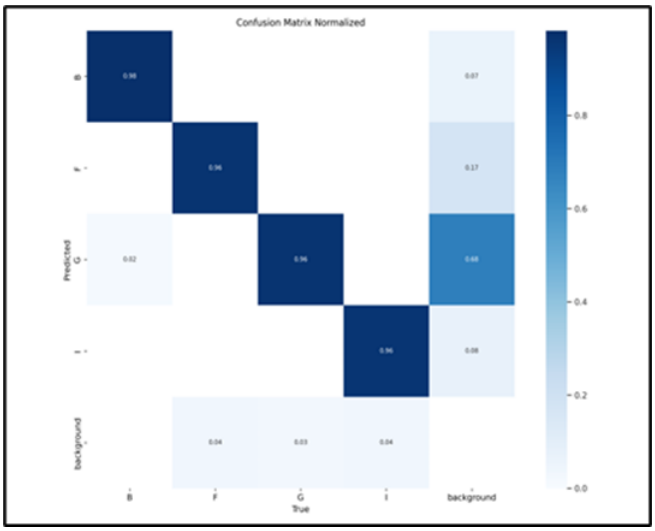

Fig. 5. Confusion matrix.

Fig. 6. Predicted result showing class G bounding box overlap.

Figs. 7 and 8 present example images with predicted defect annotations. For clarity, classes G and I were excluded to emphasize the inference results for insulator defects.

Fig. 7. Predicted result for class F.

Fig. 8. Predicted result for class B.

Although the model generally detects defects correctly in the test images, certain cases remain challenging. These occur most notably when the insulator color closely matches the background or when the insulator appears small and distant. In addition, the model frequently misclassifies class I, particularly in images showing only a partial view of an insulator or where background colors closely resemble those of the insulator.

Another observation is that the model struggles to classify flashover damage when the insulator is covered with rust or dirt. A potential solution to improve prediction accuracy in these cases is to increase the number of representative training images depicting such challenging scenarios. Leveraging utility personnel with extensive experience in dataset annotation may further enhance the adequacy of training for the intended inference task.

The YOLOv10-N evaluated in this study was used without architectural modifications, unlike in previous research where performance-oriented changes were introduced. Nevertheless, the results demonstrate satisfactory performance in detecting two

defect classes (F and B). This indicates that, despite its minimal parameter count, YOLOv10-N is capable of effective insulator defect detection.

4 Conclusion

The inspection of insulators using unmanned aerial vehicles (UAVs) equipped with on-board AI applications is an emerging trend among utilities worldwide. In this study, the YOLOv10-N model, intended for such applications, was evaluated. The model achieved adequate performance in predicting two defect classes without architectural modifications. Furthermore, K-fold cross-validation produced a highly accurate model with strong generalization on the held-out dataset.

5 Recommendations

To enhance the applicability of the YOLOv10-N model for real-world insulator inspections, future work should include training on datasets incorporating additional defect classes and a wider range of insulator types (e.g., glass insulators) [11]. Furthermore, both the training and test datasets should be annotated or quality-checked by an insulator inspection expert. Finally, it is recommended that the model be deployed and evaluated on single-board computers, such as the Raspberry Pi and Jetson Nano, using UAV-acquired video footage.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Bellou, E., Pisica, I., Banitsas, K.: Real-time object detection on high-voltage power lines using an unmanned aerial vehicle (UAV). In: Proc. 58th International Universities Power Engineering Conference (UPEC 2023), pp. 1–6. IEEE, August (2023)
2. Koubaa, A., Ammar, A., Abdelkader, M., Alhabashi, Y., Ghouti, L.: AERO: AI-enabled remote sensing observation with onboard edge computing in UAVs. Remote Sensing, 15(7), 1873 (2023)
3. Bellou, E., Pisica, I., Banitsas, K.: Aerial inspection of high-voltage power lines using YOLOv8 real-time object detector. Energies, 17(11), 2535 (2024)
4. 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, 73, 1–9 (2024)
5. Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J.: YOLOv10: Real-time end-to-end object detection. In: Advances in Neural Information Processing Systems, vol. 37, pp. 107984–108011 (2024)
6. Liu, Y., Liu, D., Huang, X., Li, C.: Insulator defect detection with deep learning: A survey. IET Generation, Transmission & Distribution, 17(16), 3541–3558 (2023)

7. Fahim, F., Hasan, M.S.: Enhancing the reliability of power grids: A YOLO-based approach for insulator defect detection. e-Prime – Advances in Electrical Engineering, Electronics and Energy, 9, 100663 (2024)
8. Kulkarni, P., Shaw, T., Lewis, D.: Insulator Defect Image Dataset – Version 1.2: Documentation. EPRI, Palo Alto, CA (2020)
9. Roboflow: Insulator defect detection dataset. Available at: https://universe.roboflow.com/insulator-defect-detection/insulator-defect-detection-veowd, last accessed 2024/11/09
10. Ultralytics: K-fold cross validation with Ultralytics. Available at: https://docs.ultralytics.com/guides/kfold-cross-validation, last accessed 2024/10/02
11. Li, Z., Jiang, C., Li, Z.: An insulator location and defect detection method based on improved YOLOv8. IEEE Access (2024)