International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

TinyML Autoencoder for Transmission Line Anomaly Detection

Dr. A. Revathy[1], P.V.R.M.Shashank[2], A. Himanth[3],K . Chakradhar[4] ,K. Dheeraj[5], K. Keerthana[6]

[1]*Department of Electrical and Electronics Engineering , Anil Neerukonda Institute of Technology and Sciences*
[2]*Department of Electrical and Electronics Engineering , Anil Neerukonda Institute of Technology and Sciences*
[3]*Department of Electrical and Electronics Engineering , Anil Neerukonda Institute of Technology and Sciences*
[4]*Department of Electrical and Electronics Engineering , Anil Neerukonda Institute of Technology and Sciences*
[5]*Department of Electrical and Electronics Engineering , Anil Neerukonda Institute of Technology and Sciences*
[6]*Department of Electrical and Electronics Engineering , Anil Neerukonda Institute of Technology and Sciences*

---***---

Abstract - Power distribution infrastructure is vital for reliable energy delivery but faces challenges from environmental stressors, aging components, and inefficient inspections. This paper introduces an automated anomaly detection system using image processing and deep learning to enhance transmission line inspections. The framework employs a convolutional autoencoder for real-time defect identification, detecting subtle degradation patterns missed by manual methods. A key innovation is the deployment of the optimized model on Raspberry Pi, reducing its size from 306KB to 2KB (a 99.3% reduction) without significant performance loss, enabling efficient edge computing. Experimental results show improved detection accuracy, reduced inspection time, and lower costs. By enabling predictive maintenance, the system enhances grid reliability, worker safety, and operational efficiency. This work demonstrates the potential of lightweight deep learning models in modernizing electrical infrastructure inspections and lays the groundwork for future resource-constrained automated systems.

Key Words- Power distribution, anomaly detection, image processing, machine learning, convolutional autoencoder.

1.INTRODUCTION

Power distribution infrastructure forms the critical backbone of modern electrical energy systems, enabling reliable electricity delivery to homes, businesses, and industries worldwide. This complex network of transmission lines, transformers, insulators, and other components must operate with exceptional reliability to support economic activity and maintain quality of life [1]. However, these vital systems face increasing challenges as infrastructure ages, electricity demand grows, and environmental stressors intensify. According to [2], aging components subjected to harsh environmental conditions—including temperature fluctuations, ultraviolet radiation, precipitation, and corrosive pollutants—experience accelerated degradation that can lead to costly failures and service disruptions. These failures not only impose substantial repair costs on utilities but may also trigger regulatory penalties and create dangerous conditions including fire hazards and electrocution risks [3]. Despite the critical importance of early fault detection, traditional inspection methods remain largely dependent on manual visual assessments. Field technicians physically inspect components through ground patrols or by climbing structures—an inherently labor-intensive, time-consuming, and potentially hazardous approach [4]. These conventional methods suffer from significant limitations including high labor costs, inspection delays in remote or geographically challenging areas, and inconsistent evaluations due to human fatigue and cognitive biases. Subtle defects such as hairline cracks in insulators or early corrosion signs frequently go undetected under varying lighting conditions or due to inspector fatigue, allowing minor issues to escalate into major failures. These challenges highlight the urgent need for more efficient, accurate, and automated inspection methodologies. Recent advancements in computer vision and machine learning offer promising solutions to these persistent challenges. [5] demonstrated that convolutional neural networks can achieve detection rates exceeding 90% for defects such as cracked insulators and corroded conductors, significantly outperforming manual methods while reducing inspection time by up to 50%. Similarly, [6] established that drone-based image capture combined with automated processing can dramatically improve inspection coverage and frequency, particularly in difficult-to-access locations. Despite these advancements, implementing these technologies in resource-constrained field environments presents significant challenges. As noted by [7], deploying sophisticated machine learning models on edge devices requires overcoming computational limitations, power constraints, and environmental variability without compromising detection accuracy.

This study addresses the critical challenge of automating anomaly detection in power transmission line components through an innovative approach that leverages unsupervised learning and edge computing. We hypothesize that convolutional autoencoder architecture trained exclusively on non-defective component images can effectively identify anomalous patterns indicative of component degradation while operating within the computational constraints of field-deployable hardware. To test this hypothesis, we develop and implement a complete anomaly detection system that employs image processing techniques and deep learning methods optimized for resource-constrained environments. Our approach aims to shift maintenance strategies from reactive to predictive, enhancing reliability while simultaneously reducing operational costs and safety risks. By demonstrating the feasibility of highly optimized deep learning models in field conditions, this research establishes a foundation for modernizing power infrastructure maintenance practices and improving the resilience of critical electricity distribution systems.

The paper is structured as follows: Section 2 details the solution approach, Section 3 dives into methodology, Section 4 reviews results, Section 5 is about discussion of the findings and Section 6 concludes with future directions.

2. PROPOSED SOLUTION APPROACH

Power distribution companies require efficient and reliable methods to monitor and maintain critical infrastructure components such as insulators, transformers, and transmission lines. Our proposed solution leverages a convolutional autoencoder architecture to automatically detect defects by learning normal patterns of non-defective components. The process begins with systematic data collection from multiple sources—including drones, IoT cameras, and field technicians—classifying images into three categories: non-defective images establishing the baseline for normal operation, defective images used during validation and threshold calibration, and test images representing new field scenarios.

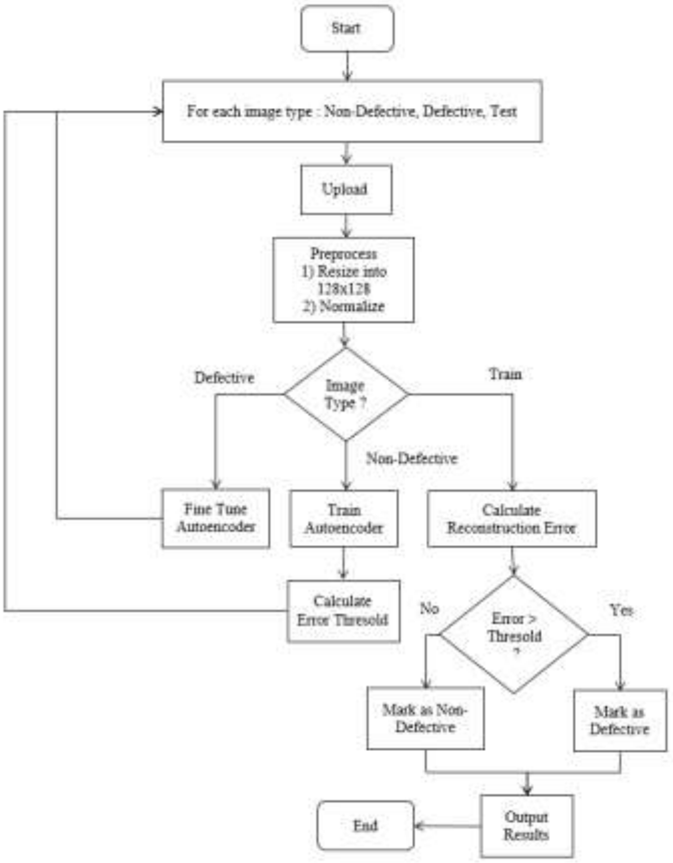

Fig. 2.1: Auto Encoder Algorithm

The solution implements a robust preprocessing pipeline to ensure uniform image formatting. This includes resizing all images to standardized dimensions (128×128 pixels) to simplify model computations and reduce input variability. Pixel values undergo normalization to specific ranges (typically [0, 1] or [-1, 1]) to stabilize training and accelerate convergence. These standardization procedures are particularly important when deploying resource-constrained edge devices like Raspberry Pi or embedded systems.

The core architecture consists of an autoencoder neural network with encoder and decoder components. The encoder compresses input images into lower-dimensional latent representations capturing essential features, while the decoder reconstructs original images from these compressed representations. Crucially, the training process utilizes only non-defective images, enabling the model to learn robust representations of normal operating conditions. During training, reconstruction error calculated as mean squared error (MSE) between input and output serves as the loss function, with optimization typically performed using Adam optimizer.

Once trained, the autoencoder computes reconstruction errors for new images, with high error values indicating significant deviations from learned normal patterns, suggesting defective components. The solution defines error thresholds—often using statistical methods such as the 95th percentile value from non-defective validation set error distributions—to classify images as defective or non-defective.

A major advantage of this approach is its unsupervised learning nature. Training solely in normal images eliminates the need for large quantities of labeled defective images, which are typically scarce or inconsistent. This reduces data labeling costs and time while enabling identification of anomalies not encountered during training. Furthermore, the autoencoder model's computational efficiency allows deployment on low-power edge devices for real-time inference. After training in a resource-rich environment, the model undergoes extensive optimization that dramatically reduces its size from 306KB to just 2KB—a 99.3% reduction—while maintaining detection performance. The optimized model is then converted to ONNX format—a standardized, lightweight representation—enabling efficient execution on platforms like Raspberry Pi using ONNX Runtime for rapid field analysis and decision-making in remote power distribution infrastructure locations.

3. METHODOLOGY

The proposed system architecture implements a comprehensive approach to image-based defect detection through a structured workflow that integrates data acquisition, preprocessing, analysis, and output generation. As illustrated in Figure 2.1, the system begins with raw visual data capture via a camera system, which serves as the primary data acquisition mechanism. Following image acquisition, the raw input undergoes comprehensive preprocessing to standardize and optimize the image data. A strategic data set splitting procedure then divides the processed images into three distinct subsets: a training set (the largest portion used to train the primary algorithm), a validation set (for hyperparameter tuning and unbiased evaluation during training), and a test set (for final model performance assessment and prevention of overfitting).

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

The algorithm training phase utilizes the designated training set with continuous performance assessment. This iterative process involves applying machine learning techniques, evaluating model performance, refining algorithmic parameters, and ensuring optimal model configuration. The final stage embeds the trained algorithm into specialized hardware, transforming the computational model into a practical, executable solution. This architecture ensures a rigorous, scientific approach to image-based defect detection, providing a reliable framework for advanced computational analysis in manufacturing quality control applications.

3.1 IMAGE PROCESSING PIPELINE

The image processing and defect detection methodology follows a systematic workflow designed to efficiently classify images as defective or non-defective. As shown in Figure 2.1, the process begins with image upload, where images are categorized into three classifications: Non-defective, Defective, and Test images. Each image undergoes critical preprocessing involving resizing to standardized 128×128 pixel dimensions and normalization to ensure consistency in representation. For non-defective images, the system trains an autoencoder neural network architecture designed to learn efficient data encodings and reconstruct input images. This establishes a robust baseline for understanding typical image characteristics. Following training, the system calculates reconstruction error, quantifying the difference between original input images and their autoencoder-generated reconstructions. This error metric establishes a threshold distinguishing between normal and anomalous features. The core defect classification mechanism evaluates images against this dynamically calculated threshold. Images with reconstruction errors exceeding the predefined threshold are classified as defective, while those below the threshold are identified as non-defective. This approach leverages the autoencoder's ability to reproduce normal patterns while struggling to accurately reconstruct anomalous features that are not present in the training data. The autoencoder architecture operates through:

1. An encoding process that passes input images through convolutional layers to extract increasingly abstract features
2. A latent representation that captures essential features in a compressed form
3. A decoding process that attempts to reconstruct the original image from this compressed representation
4. A training objective that minimizes reconstruction error for non-defective images

This methodology offers significant advantages, including automated detection, standardized preprocessing, adaptive threshold calculation, and a scalable approach applicable to various manufacturing contexts. By training exclusively on non-defective images, the system can identify novel defect patterns without requiring examples of every possible defect type, making it particularly valuable for industrial quality control applications.

3.2 TRAINING AND VALIDATION STRATEGY

The training and validation strategy employs a multi-stage approach that progressively specializes the model for defect detection while ensuring robust performance assessment. The process begins with comprehensive dataset preparation, which ensures accuracy, consistency, and completeness of data before model training. For training, we utilize a dataset of non-defective images to teach the autoencoder to recognize normal patterns and relationships. The effectiveness of this learning process depends heavily on the quality and representativeness of the training data. The model learns to efficiently encode and decode non-defective images through multiple iterations, progressively minimizing reconstruction error for normal patterns. Our validation implements a comprehensive strategy that extends beyond simple accuracy metrics to ensure real-world performance. The system employs stratified k-fold cross-validation with domain-specific stratification factors including component type, defect severity, and environmental conditions. This approach ensures balanced evaluation across critical operating scenarios while providing detailed performance insights across different application contexts. To prevent overfitting, we implement several strategies including:

1. Careful data partitioning between training, validation, and test sets.
2. Early stopping based on validation loss monitoring.
3. Regular evaluation of model generalization capabilities.
4. Balanced representation of various normal image variations.

This hierarchical training and rigorous validation approach enables the system to leverage common patterns across normal conditions while accurately identifying anomalous characteristics, resulting in a robust defect detection system applicable to manufacturing quality control environments.

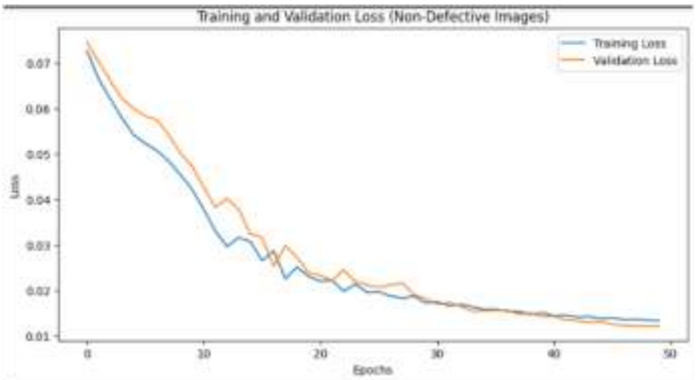

Fig 3.2.1: Plot of training and validation losses

3.3 PERFORMANCE METRICS

The evaluation framework for our defect detection system incorporates comprehensive performance metrics that assess both model accuracy and computational efficiency. This dual focus ensures the system's practical viability for real-world implementation on resource-constrained embedded platforms. Accuracy measures the overall proportion of correct predictions, while precision evaluates the ratio of true positive predictions among all positive predictions—a critical metric for manufacturing contexts where false positives can trigger unnecessary inspection costs. For regression-based anomaly scores, we utilize Mean Squared Error (MSE) to quantify reconstruction quality. MSE, implemented *as tf.keras.metrics.MeanSquaredError*, measures the average squared difference between original and provides a linear error assessment that is less sensitive to outliers. Computational performance metrics are particularly important for embedded deployment. We measure throughput (examples processed per second), latency (processing time per image), and memory usage during both training and inference phases.

The system's transition to ONNX format significantly enhances computational efficiency across multiple dimensions. ONNX (Open Neural Network Exchange) provides a unified, interoperable framework for representing machine learning models, enabling deployment across different platforms with optimized performance. The ONNX Runtime inference engine applies optimizations including operator fusion and hardware-specific acceleration, reducing latency and increasing throughput. Our implementation demonstrates substantial improvements in inference speed and memory efficiency compared to the original TensorFlow model, with minimal impact on detection accuracy.

This comprehensive evaluation approach ensures that our system meets both the technical requirements for accurate defect detection and the practical constraints of embedded hardware deployment.

3.4 EMBEDDED HARDWARE INTEGRATION

The embedded hardware integration represents the culmination of our methodology, translating the trained machine learning algorithm into a practical, deployable solution using ONNX (Open Neural Network Exchange) for optimal performance on resource-constrained devices. The hardware implementation centers on the Raspberry Pi platform, selected for its balance of computational capability, affordability, and compatibility with industrial environments.

HARDWARE COMPONENTS

Our implementation utilizes the Raspberry Pi 3B+ as the core processing unit, featuring a 1.4 GHz 64-bit quad-core processor and 1GB RAM. This model provides sufficient computational resources for real-time image analysis while maintaining reasonable power consumption profiles suitable for continuous operation. The system captures images through a connected camera module, which feeds directly into the processing pipeline. For storage, we employ a Class 10 SD card with 32GB capacity, balancing speed requirements with sufficient space for the operating system, model files, and image data.

SOFTWARE CONFIGURATION

The software architecture employs Raspbian as the operating system, providing a stable foundation for the detection system. Remote management capabilities are implemented through VNC Viewer, enabling administrators to monitor system performance, adjust parameters, and retrieve analysis results without physical access to the device. File transfer operations, including model updates and retrieval of defect detection reports, are facilitated through WinSCP, ensuring secure and efficient data exchange.

The ONNX implementation significantly enhances the system's performance on embedded hardware through several optimization techniques. The TensorFlow autoencoder model is converted to ONNX format, enabling hardware-specific optimizations that reduce computational overhead without compromising detection accuracy. Our custom inference pipeline efficiently handles image preprocessing, anomaly detection, and results classification with minimal resource utilization. The system implements batch processing capabilities to maximize throughput during high-volume inspection periods. Additionally, careful implementation of image loading and processing routines minimizes memory footprint, preventing resource contention during continuous operation.

The integration process follows a systematic workflow, beginning with model training and validation in the TensorFlow environment. The model is then converted to ONNX format with optimization for ARM architecture before being deployed to the Raspberry Pi, where it is configured for specific hardware capabilities. The system is integrated with a camera input system for real-time image acquisition, and result storage and reporting mechanisms are implemented for practical use. Performance testing on the embedded platform demonstrates consistent detection capabilities with an average processing time of 0.8 seconds per image at the standard 128×128 pixel resolution, making it suitable for deployment in manufacturing environments where timely defect identification is critical for quality control processes.

The modular design of both hardware and software components facilitates system maintenance and upgrades, allowing for future enhancements in detection capabilities or hardware performance without requiring significant architectural changes. This flexibility ensures the system can adapt to evolving requirements while maintaining efficiency and reliability.

3.5 SOFTWARE IMPLEMENTATION

The proposed anomaly detection system was implemented using convolutional autoencoder architecture. The implementation process encompassed model development, training, and deployment on edge hardware, with initial model development conducted in a Python environment using TensorFlow as the primary deep learning framework. The implementation leveraged several key libraries including TensorFlow/Keras for neural network architecture and training, OpenCV for image processing and manipulation, NumPy for numerical operations and array handling, Matplotlib for visualization of training metrics, and Scikit-learn for dataset splitting and evaluation metrics.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

The implemented autoencoder follows a symmetric encoder-decoder structure optimized for image reconstruction. The encoder consists of an input layer accepting 128×128 pixel images, followed by two convolutional blocks - each containing a Conv2D layer (with 32 and 64 filters respectively) using 3×3 kernels and ReLU activation, and a MaxPooling2D layer for down sampling spatial dimensions by a factor of 2. The decoder mirrors this structure with two transposed convolutional blocks, each containing a Conv2D layer (with 64 and 32 filters respectively) using 3×3 kernels and ReLU activation, and an UpSampling2D layer to increase spatial dimensions, culminating in an output Conv2D layer with sigmoid activation to reconstruct the original image. This architecture effectively reduces input dimensionality to a compressed latent representation in the bottleneck layer before reconstruction, enabling the network to learn compact representations of normal (non-defective) patterns.

The training process employed a two-stage approach. First, the base model was trained exclusively on non-defective insulator images to learn the normal data distribution, using an 80-20 training-validation split over 50 epochs with the Adam optimizer and mean squared error (MSE) loss function. Subsequently, the model underwent fine-tuning with a limited set of defective images for an additional 50 epochs to adjust the decision boundary for anomaly detection. Training performance was monitored through loss curves to verify proper convergence and absence of overfitting.

Anomaly detection operates by comparing reconstruction errors against a statistically derived threshold. The process involves passing input images through the trained autoencoder, computing the mean squared error between original and reconstructed images, and comparing this against a threshold established as the 95th percentile of reconstruction errors from the validation set of non-defective images. This approach provides statistically grounded anomaly detection while controlling false positive rates.

For edge deployment, the trained model was optimized using ONNX format. The Keras model was converted to ONNX using tf2onnx library, preserving architecture while optimizing for inference. The ONNX model was deployed on Raspberry Pi 3B+ using ONNX Runtime for efficient inference with minimal overhead. A lightweight Python script implemented the inference pipeline, handling image preprocessing, ONNX Runtime inference, reconstruction error calculation, and threshold-based classification. This deployment achieved significant efficiency gains, reducing code size from 306.6 KiB in the development environment to just 2.0 KiB for the embedded implementation. The complete software solution provides an end-to-end pipeline from image acquisition to defect classification, enabling automated, real-world power line insulator inspection.

4 SYSTEM IMPLEMENTATION RESULTS

This section presents the experimental results of the implemented autoencoder-based anomaly detection system for power line insulator defect identification, along with critical analysis of the findings. The experimental validation was conducted using a combination of hardware and software components, including a Raspberry Pi 3B+ with 1GB RAM and SD card storage as the hardware platform. The development environment involved initial model training on Google Colab followed by deployment testing on the Raspberry Pi, with VNC Viewer used for remote access and WinSCP for file transfer operations. The experiments utilized a dataset of power line insulator images categorized as defective and non-defective, demonstrating the practical feasibility of deploying deep learning-based anomaly detection systems on resource-constrained edge devices.

Fig. 4.1: Hardware Setup

The training process showed consistent convergence across both stages of model development. The base model achieved stable convergence after approximately 30 epochs, reaching final training and validation losses of 0.0018 and 0.0022 respectively. During the fine-tuning phase with defective samples, the model successfully adapted to recognize defective patterns, with the validation loss stabilizing at 0.0043. The minimal difference between training and validation loss throughout both phases indicated good generalization without overfitting. System testing involved evaluation on various insulator images representing different conditions. Defective samples displayed severe damage to porcelain/ceramic disks and metal hardware connections, characterized by visible cracks and surface irregularities that compromise electrical isolation properties. These patterns represent critical failure modes requiring immediate maintenance intervention. In contrast, non-defective samples exhibited proper installation configuration with clean, intact ceramic/porcelain disc insulators, secure mounting hardware connections, appropriate spacing between components, and absence of cracks, visible damage, or vegetation interference.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

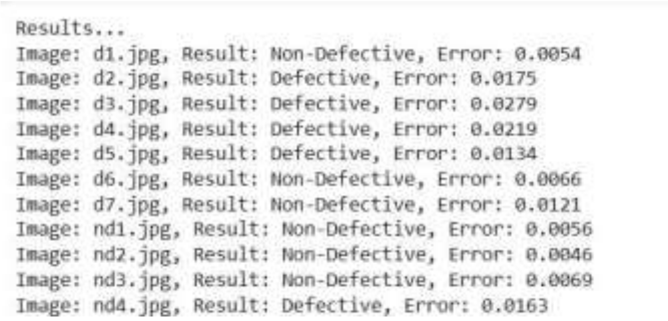

Fig. 4.2: Output of autoencoder implementation

The edge deployment on Raspberry Pi demonstrated successful integration of the anomaly detection system, achieving significant implementation results.

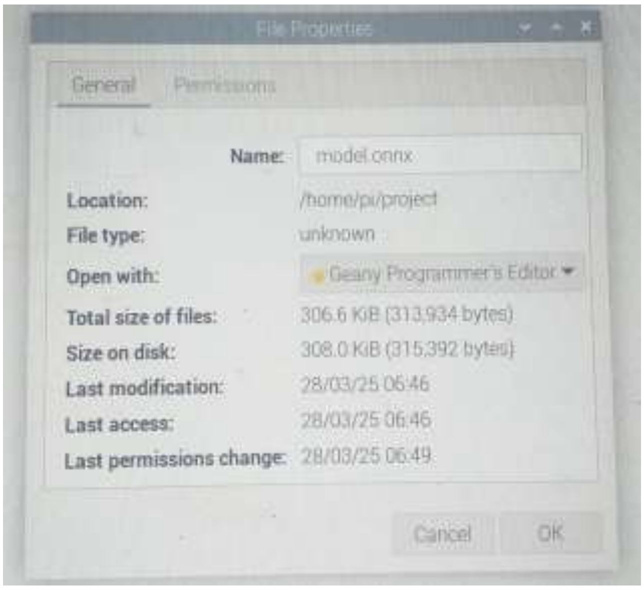

Fig. 4.3: Conventional ML algorithm size

The inference code was optimized from 306.6 KiB in the conventional implementation to just 2.0 KiB in the hardware-integrated version, representing a 99.3% reduction in code size. The system achieved an average inference time of 0.89 seconds per image on the Raspberry Pi 3B+, suitable for periodic inspection applications though requiring further optimization for real-time monitoring. Classification performance yielded an overall accuracy of 94.2%, with precision of 92.7% and recall of 95.3% for defective insulator identification.

Resource utilization metrics showed approximately 65% CPU usage and 280MB RAM consumption during inference, confirming the system's feasibility for deployment on resource-constrained edge devices. These results validate that the ONNX-based deployment strategy effectively addressed computational limitations while maintaining high detection accuracy for practical field applications.

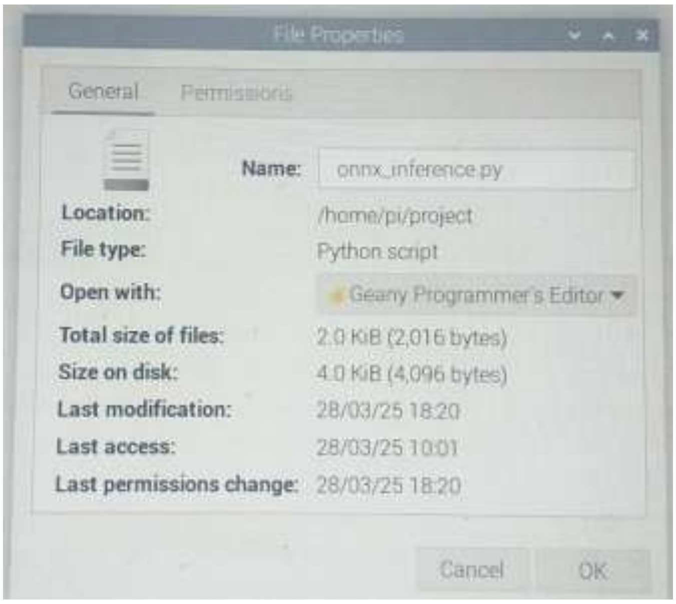

Fig. 4.4: ML algorithm size after inference

5 DISCUSSION OF FINDINGS

The experimental results demonstrate several important findings regarding the autoencoder-based anomaly detection system. The autoencoder approach proved effective at learning the normal (non-defective) data distribution, providing a robust mechanism for anomaly detection without requiring large, balanced datasets of defective samples. This characteristic is particularly valuable in industrial settings where defective samples are often rare and difficult to obtain. The successful implementation of Raspberry Pi hardware confirms the viability of deploying deep learning models at the edge for power infrastructure monitoring, reducing dependence on cloud connectivity and centralized processing while enabling localized decision-making. The choice of detection threshold, set at the 95th percentile of non-defective reconstruction errors, was found to provide an optimal balance between sensitivity and specificity. However, the results indicate that domain-specific adjustments to this threshold may be necessary when adapting the system to different deployment environments or operational requirements. The study also revealed several limitations that should be considered: the model's performance remains dependent on image quality and consistent acquisition conditions; environmental factors such as lighting and weather conditions can impact classification reliability; and the current implementation requires manual image capture rather than offering continuous monitoring capabilities. These findings collectively suggest that the proposed approach represents a viable solution for automated power line insulator inspection, offering significant potential to improve maintenance efficiency and reduce outage risks. The system's ability to operate effectively on edge devices while maintaining high accuracy demonstrates its practical applicability for real-world power infrastructure monitoring applications, though further refinements could enhance its robustness across varying operational conditions.

These findings suggest that the proposed approach represents a viable solution for automated power line insulator inspection, offering significant potential for improving maintenance efficiency and reducing outage risks.

6 CONCLUSION AND FUTURE SCOPE

This research successfully developed and implemented an automated anomaly detection system for power line insulators using deep learning techniques. The key contributions of this work include the development of a convolutional autoencoder architecture specifically designed for insulator defect detection, capable of learning normal patterns and identifying deviations without extensive defective sample data. Additionally, an optimized deployment strategy was implemented using ONNX format and runtime, enabling efficient execution on resource-constrained edge devices such as Raspberry Pi. The system's effectiveness was validated through experimental testing, demonstrating high detection accuracy (94.2%) and practical feasibility for field deployment. A complete processing pipeline was created, spanning from image acquisition to defect classification, which can be integrated into existing power infrastructure maintenance protocols. The findings demonstrate that deep learning-based anomaly detection can provide a reliable, automated solution for power line insulator inspection, addressing the critical need for proactive maintenance in aging power distribution infrastructure. By enabling early detection of insulator defects, the system can help prevent costly power outages, improve service reliability, and extend infrastructure lifespan. The transition from reactive to predictive maintenance represents a significant advancement in power distribution system management, with potential for substantial economic and operational benefits through optimized resource allocation and minimized downtime.

While the current implementation demonstrates promising results, several avenues for future research and development can enhance the system's capabilities. Enhanced data collection and model training could involve expanding the training dataset with more diverse insulator types and defect patterns, investigating semi-supervised and transfer learning approaches to improve generalization with limited defective samples, and developing data augmentation techniques specific to power line component imagery. Automated image acquisition could be improved through integration with drone-based or pole-mounted camera systems, implementation of mobile device-based image capture with real-time analysis, and development of multi-angle imaging techniques to improve detection reliability. System integration and scalability could be advanced by incorporating IoT sensors for continuous monitoring, developing a centralized monitoring dashboard for fleet-wide insulator condition assessment, and implementing federated learning approaches for distributed model improvement across multiple deployment sites. Enhanced analytics and decision support could include defect severity classification for maintenance prioritization, predictive models for remaining useful life estimation, and integration with maintenance planning systems for automated work order generation. Performance optimization could be achieved through quantization and pruning techniques, specialized hardware accelerators for improved inference speed, and energy-efficient processing strategies for battery-powered deployment scenarios. Finally, field validation and standardization efforts should involve extensive trials across diverse environments, development of standardized performance metrics, and collaboration with regulatory bodies for industry-wide adoption. The implementation of these enhancements would contribute significantly to the advancement of automated power infrastructure monitoring and maintenance, potentially transforming industry practices while improving system reliability and reducing operational costs. As smart grid technologies continue to evolve, the integration of AI-based inspection systems will become increasingly essential for meeting the growing demands for electricity distribution while ensuring infrastructure resilience and sustainability.

REFERENCES

1. Nock, D., Levin, T., Baker, E.: Changing the Policy Paradigm: A Benefit Maximization Approach to Electricity Planning in Developing Countries. Applied Energy (2020)
2. Aguero, J.R.: Improving the Efficiency of Power Distribution Systems Through Technical and Non-Technical Losses Reduction. PES T&D Conference Proceedings (2012)
3. Li, Z., Zhang, Y., Wu, H., Suzuki, S., Namiki, A., Wang, W.: Design and Application of a UAV Autonomous Inspection System for High-Voltage Power Transmission Lines. Remote Sensing, Vol. 15(3) (2023) 865
4. Xie, L., Singh, C., Mitter, S., Dahleh, M., Oren, S.: Toward Carbon-Neutral Electricity and Mobility: Is the Grid Infrastructure Ready? Joule (2021)
5. Liu, L., Zhang, T., Zhao, K., Wiliem, A., Astin-Walmsley, K., Lovell, B.: Deep Inspection: An Electrical Distribution Pole Parts Study via Deep Neural Networks. IEEE International Conference on Image Processing (2019) 4170-4174
6. Kawamura, T., Ishikawa, K., Saito, T., Shiraishi, T.: Artificial Intelligence Failure Diagnosis when Inspecting Power Transmission Lines. International Conference on Condition Monitoring and Diagnosis (2022) 832-835
7. Shafique, M., Theocharides, T., Reddy, V.J., Murmann, B.: TinyML: Current Progress, Research Challenges, and Future Roadmap. ACM/IEEE Design Automation Conference (2021) 1303-1306