


RESEARCH

Open Access



# Deep learning for component fault detection in electricity transmission lines

Ilyke Maduako<sup>1,2\*</sup> , Chukwuemeka Fortune Igwe<sup>3</sup>, James Edebo Abah<sup>1</sup>, Obianuju Esther Onwuasaanya<sup>1</sup>, Grace Amarachi Chukwu<sup>1</sup>, Franklin Ezeji<sup>4</sup> and Francis Ifeanyi Okeke<sup>1</sup>

\*Correspondence:  
lkechukwu.maduako@unn.edu.ng;  
imaduako@unicef.org

<sup>1</sup> Office of Innovation, UNICEF,  
New York, NY 10017, USA

<sup>2</sup> University of Nigeria Enugu  
Campus, Nsukka, Enugu State,  
Nigeria

<sup>3</sup> New University of Lisbon,  
Lisbon, Portugal

<sup>4</sup> Federal University  
of Technology, Owerri, Imo State,  
Nigeria

## Abstract

Component fault detection and inventory are one of the most significant bottlenecks facing the electricity transmission and distribution utility establishments especially in developing countries for delivery of efficient services to the customers and to ensure proper asset audit and management for network optimization and load forecasting. For lack of technology and data, insecurity, the complexity associated with traditional methods, untimeliness, and general human cost, electricity assets monitoring, and management have remained a big problem in many developing countries. In view of this, we explored the use of oblique UAV imagery with high spatial resolution and fine-tuned deep Convolutional Neural Networks (CNNs) for automatic faulty component inspection and inventory in an Electric power transmission network (EPTN). This study investigated the capability of the Single Shot Multibox Detector (SSD), a one-stage object detection model on the electric transmission power line imagery to localize, detect and classify faults. Our proposed neural network model is a CNN based on a multiscale layer feature pyramid network (FPN) using aerial image patches and ground truth to localise and detect faults through a one-phase procedure. The SSD Rest50 architecture variation performed the best with a mean Average Precision (mAP) of 89.61%. All the developed SSD-based models achieve a high precision rate and low recall rate in detecting faulty components, thus achieving acceptable balance levels of F1-score and representation. We have established in this paper that combined use of UAV imagery and computer vision presents a low-cost method for easy and timely electricity asset inventory, especially in developing countries. This study also provides the guide to various considerations when adopting this technology in terms of the choice of deep learning architecture, adequate training samples over multiple fault characteristics, effects of data augmentation, and balancing of intra-class heterogeneity.

**Keywords:** Power-line fault detection, Deep learning, UAV imagery, Single Shot Multibox Detector

## Introduction

Recently, Nigeria's development agenda has been anchored in a vision that identifies energy as one of the vital infrastructural enablers for development. With the realisation that for them to successively make a significant positive transition in development, there

must be an efficient, reliable, vast, and environment-friendly energy source, transmission, and distribution. This means that majority of the burden of energy demand is on power companies to provide and transmit quality energy services to consumers. Against this backdrop, investors in their transmission lines need accurate, cost and time-efficient methods to carry out existing asset inventory of the transmission lines for well-informed decisions and investment. In view of this, our hypothesis is that deep learning on high-resolution Aerial images should provide a cost and time-effective solution for power line assets inventory and studies, study fields.

Regular inspection of electric power lines has become an essential concern because virtually all human activities, infrastructure services, and businesses will collapse without electricity [1]. Generally, in many developing countries, the available electricity is unreliably characterised by households and businesses experiencing long and frequent power outages resulting from electricity demand exceeding available electricity supply caused by load shedding and/or technical failures [2, 3]. For example, Electric Utilities in Nigeria claimed that: some sections of the grid are outdated with inadequate redundancies; regular vandalization of the lines associated with a low level of surveillance and security on all electrical infrastructure, and the serious lack of required modern technologies for communication and monitoring is causing more and more power outages [4, 5]. To tackle these challenges, there are different approaches that have been developed for fault detection on the power transmission lines. Among these methods is the use of Machine Learning techniques on Very High Resolution (VHR) satellite imagery. This method has proven to be more efficient and outperform manual inspection and traditional data analysis approaches for detecting faults in power transmission lines at large.

Remote sensing techniques have been very efficient in power line corrosion and mechanical loss detection. Inspection of the electricity power transmission network (EPTN) especially in remote areas using remote sensing techniques requires very high-resolution images such as those obtained from aerial surveys, UAV images, and Lidar point clouds data. Unmanned Aerial Vehicle (UAV) surveillance has become the state-of-the-art in power line inspection for defects and damage [6]. Many studies have also demonstrated the efficacy of high-resolution remote sensing techniques in power line inspection and monitoring. For example, Xue et al. [7], utilized SAR imagery to measure electricity towers' damage caused by landslides. The use of high-resolution TerraSAR-X imagery to track power line damages in natural disaster situations has also been discussed in [8–10].

Studies using optical remote sensing have focused on fault diagnosis for the different EPTN components themselves because the ground sample data (GSD) is usually less than the individual components' size, especially for those caused by the adjoining environment. As a result, most power line inspection studies using optical remote sensing are fixated on vegetation encroachment and minimum height and clearance distance [11–13]. Apart from vegetation encroachment, a variety of papers addressed automatic inspection of insulators' condition. These techniques aimed to take images of the insulators periodically and use automated classification methods to identify damaged insulators. Reddy et al. [14], for example, used fixed cameras on poles. Jiang et al. [15], using a photogrammetric method, addressed flashover faults—pollution-related flashes affecting insulators. In the experiment, a sensing camera placed on a tripod was

used. However, most remote optical sensing techniques are primarily restricted by the atmosphere.

Despite extensive studies on powerline inspection and fault detection, the advantages of using remote sensing in sub-Saharan Africa remain unseen due to the data unavailability and peculiarity of the power line in this region. Many utility companies and investors rely on poorly collected data from ground-based surveys, multispectral visible colour images, and most recently video surveillance of transmission line fault inspection and monitoring [16]. UAV monitoring offers high-spatial multispectral images that deal with the limitation of other remote sensing methods because of the ability to capture accurate images of transmissions components at closer proximity [17]. UAVs are able to detect small-scale defects such as broken fittings and missing knobs and can be incorporated with other modes of remote sensing. In comparison to manual methods, with limited resources and man-hours, inspecting and monitoring long transmission line corridors for potential faults and damages becomes almost impossible.

For cost and time-effectiveness, electricity infrastructure inspection, and fault diagnosis especially in transmission lines, the combination of UAV data and deep-learning techniques is imperative [6]. The advent of deep learning, which uses not only spectral information but also spatial, topological, spectral, and geometric properties of objects in images, is at the forefront of these developments. Deep learning has demonstrated potential promising advances in power line extraction and other study fields. Currently, improved algorithms and multilayer networks such as Convolution Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and reinforcement learning have demonstrated more outstanding performance than standard approaches, particularly in power line identification, transmission components detection, as well as in, vegetation encroachment prevention [18]. Conversely, the traditional approach for pattern recognition depends on the continuous engineering of parameters that are well built by humans. Hence, making the manual extraction process inefficient, unfavourable, inadequate for generalization necessities, and time-consuming. With deep learning algorithms, visual perception to extract feature hierarchies and generalization ability is enhanced on several levels [19]. These algorithms have demonstrated that conventional learning methods are sluggish and unreliable; they require substantial post-processing attempts to differentiate between transmission infrastructure [20]. Succinctly, power transmission network mapping and fault inspection require a more advanced adequate hybrid classifier that is way beyond task-based approaches, promoting the improved performance of visual recognition tasks and successfully adapts learning from multimodal data sensors for object detection.

1. Taking all these considerations into account, the use of deep learning technologies, together with the advantages of Unmanned Aerial Vehicles (UAVs), can play a fundamental role in relieving the current limitations of traditional power transmission monitoring, which are mainly based on manual operations such as electricity pole climbing, foot patrols, vehicle inspections, and field verification reports [21, 22]. Nevertheless, the operational applicability of Deep learning techniques on UAV surveillance has not yet been addressed in the context of power infrastructure inspection in developing countries, which is precisely the gap that motivates this work. This

paper explores the use of deep learning techniques for power line fault detection and inspection from UAV imagery through the lens of a case study in Nigeria. The major contributions of this paper are: the development of a single-phase deep learning model for power line faulty component detection and classification pipeline for a series of faults (multiclass) that typically exists in power transmission components. Based on the comprehensive review work in [18], very little research has been carried out in this area with the context of helping investors in a developing country like Nigeria for time and cost-effective EPTN inventor [18].

2. Exploring the feasibility of low-cost drone equipment to monitor electricity transmission infrastructure for faulty-component detection.
3. Empirical and comparative analysis of hyperparameters in CNN backbone architectures consisting of more than one electricity power line component-fault type to evaluate the effectiveness of our proposed approach.

### Related work

Most widely used models in literature for fault detection in power line components primarily involve clustering, mathematical-based techniques such as Hough transform, Gabon filters, knowledge-based techniques, and traditional pattern recognition techniques or low-level filters. For instance, in detecting broken transmission line spacers, a Canny edge detector combined with Hough transform was used by Song et al. [23]. First, a scan window was formed in the path of the conductor and during the convolution process, if there are a candidate's spacers, they are recognized in all sliding windows. Finally, the shape configuration parameter is structured to decide whether the sensed spacer was broken based on the measurement of linked parts. A study by Zhai et al. [24] exploited a pattern descriptor (variable) using the Saliency Aggregating Faster Pixel-wise Image (FPISA) for insulator extraction. Based on the colour channel in the Lab colour space, the observed insulator's flashover region was extracted. The system was tested using 100 flashover fault insulating images and obtained a detection rate of 92.7%. Utilising a similar concept, Zhai et al. [25] and Han et al. [26] detected the faults associated with the missing cap of insulators based on saliency and adaptive morphology (S-AM), a combination of shape and pattern parameters. Concentrating on the merits of the preceding investigations, while demonstrably great accuracies were acquired, especially for single EPTN components; however, these approaches are cumbersome, time-consuming, and the processing of the data necessitates a certain degree of competence. Additionally, the methodology is insufficient for multi-class classification, individual location, and identification of faults, particularly in complicated natural environments, as is the case with EPTN components.

Alternatively, drones are becoming more prevalent, and their adoption enabling remote sensing is becoming increasingly desirable in places with technological limits, owing to their low cost and rapid deployment in a variety of settings. UAVs can be equipped with increasingly sophisticated sensors such as hyperspectral (HS) and LiDAR to distinguish ETPN components from their associated faults, and the combination of these sensors has resulted in high accuracy. However, these sensors are typically quite expensive, and data processing requires a certain amount of skill and computing

capacity that is not always available. Recent advancements in drone technology and low-cost sensors have cemented a critical role for UAV monitoring of EPTN components. Furthermore, experts express that the most challenging type of faults to detect in power transmission lines are faults on EPTN components appearing as tiny aspect ratio in the captured images, for instance, power line fitting such as missing pin, nut, bolts, and a small degree of fault severity on some large components.

Since the majority of drones now include a standard high-resolution RGB camera, utilising these cameras for minuscule EPTN fault identification is advantageous in places where EPTN component overloading is an issue but funding for more advanced sensors is not easily available, and maintenance is heavily required. To detect such faults, aerial images are captured with the RGB cameras close to the exact components containing the faults or the components (or faults) cropped from the original image manually [22], automatically, or via segmentation [27]. Fu et al. [28] implemented a dynamic model for the missing pin type of faults. The fitting is a combination of multiple sections, which include pin and nut. The haar-like attribute and Adaboost classifier was used to detect each part of the fitting. The methodology involved first extracting the segmented region and circles with LSD and Hough transform, respectively, to identify the missing pin. The missing pin fault was finally obtained and then observed based on the distance limit between the centre of the circle and the pin section. Other methods in particular machine learning methods majorly Adaboost [28], and SVM [29] have been applied to an abundance of imagery for the automatic identification and monitoring of the electric power transmission network (EPTN) component faults to augment and infer additional information in recent years. These techniques have contributed to the detection of EPTN components faults with successful results. However, since they often include additional feature engineering, they become less attractive, especially for the RGB drone imagery dataset.

With the introduction of computer vision, the limitations of RGB cameras and traditional approaches to detecting and classifying EPTN faults are gradually being addressed. One of the earliest works on fault detection using deep learning was detecting surface discoloration due to flashover on an insulator using CNN classifier with pre-trained AlexNet published by Zhao et al. [30]. The experiments achieved a score of 98.71% mean Average Precision (mAP) on 1000 samples. The proposed architecture outperformed the conventional handcrafted approach but was limited to just insulator condition inspection image classification, which demanded significant feature engineering. Additionally, Faster R-CNN was presented by Liu et al. [31] to identify insulators with missing caps. The system was tested for three different voltage transmission line levels with 1000 training samples and 500 research samples prepared for each level. About 120 photographs (80 for training) were used to test the diagnosis of missing cap faults. The study also highlights the possibility of overfitting due to the small dataset and employs data augmentation to physically extend the dataset. To handle similar faults as in [31] across multi-scale level drone imagery, Jiang et al. [32] developed a novel approach using SSD as the meta-architecture for multi-level perception (low, mid, and high perception) based on ensemble learning to extract the missing insulator fault from the image resolution of  $1920 \times 1080$ -pixel. The middle and high-level perception images are made via the Region of Interests (ROIs) Union Extraction (RUE) image pre-processing. The proposed

approach's absolute precision and recall rates were 93.69% and 91.23% respectively on the test image dataset with various perception levels containing missing cap insulator problems. However, these papers considered the contextual characteristics of one type of fault inspection that affect the insulator component across the transmission corridor neglecting other defects that coexist. In most cases, the features derived by such methods may not adequately reflect the insulators, and these approaches may need the imagery modified.

Generally, convolutional layers have been shown to reliably predict specific EPTN component faults using more than just spectral indices. Deep learning successfully incorporates additional criteria such as shape and texture (semantic representations) to provide more accurate predictions about EPTN component faults. Nonetheless, one particular issue in power line fault detection using deep learning CNN is data insufficiency. This is because the DL model is required to generalize the solution at the end of the training. To achieve this, a robust and large amount of dataset is usually required. In the previous papers to circumvent potential data shortages and expedite the creation of a reliable predictive model, attempts were made to synthesize the images (e.g., in [33]) and data augmentation (e.g., in [31, 34]). Other researchers have examined the use of transfer learning and few-shot learning to identify fault types. For instance, for lack of sufficient training images, Bai et al. [35] utilized a transfer learning process using the ImageNet data kit, which included a 1.2 million samples dataset. This model was then trained, i.e., fine-tuned by the limited data set acquired containing the surface fault of insulators based on the Spatial Pyramid Pooling networks (SPP-Net) with transfer learning approaches. This allowed the weight optimization to begin at the top layers (where there is a different feature complexity from the original training data utilized) in the 3D CNN of the SPP-Net adopted rather than for the whole model. The result showed the better performance of SPP-Net architecture with transfer learning over the RGB imagery in a short computation time. Although this model proved sufficient, the result was limited to a classification problem involving just the insulator fault.

In recent years, there have been few efforts to develop a deep learning approach, to identify several power lines faults simultaneously. Typically, a two-step object detection technique is commonly utilized: first, to identify the component, and second, to detect the fault in those components. In this light, Tao et al. [33] developed two separate backbone models, Defect Detector Network (DDN) DDN and Insulator localizer Network (ILN) based on the Visual Geometry Group (VGG) model and Residual Network (ResNet) model respectively, on the domain knowledge of the EPTN component's structure. To find a missing cap fault, a cascading architecture combining a custom-developed ILN and a DDN model was utilized. The ILN identifies all the insulators in the aerial image and then cuts the detected areas and feeds them into the DDN. A total of 900 regular images were collected from UAVs for this experiment and 60 defective images. Data insufficiency was tackled by segmenting the image using the U-net algorithm to divide the output of the ILN into insulator and background. The segmented insulator was then combined with distinct images of different backgrounds to mimic real-life background situations concerning insulator position. The result of this was then merged as input for the DDN model. Finally, about 1956 pictures for ILN (1186 for training) and 1056 images with missing caps (782 for training) were prepared. The DDN detection precision and



recall are 0.91 and 0.96. The resulting accuracy outperformed the direct use of existing frameworks. However, most related studies do not consider a single-phase approach and do not detect more than one fault simultaneously, but rather focus on video surveillance and single class fault detection on the transmission lines. Exploring the performance of different object detection deep learning models, the SSD meta-architecture utilized by Jiang et al. [32] performed well considering the multiscale camera imagery perception and model characteristics.

Despite the capacity to exploit additional object attributes for object detection and classification prediction, [32] demonstrated how such approaches have limitations. They propose additional research into CNN models and their performance in landscapes with a variety of vegetation patterns, complex backgrounds, and barriers. The trade-off for increasing the spatial resolution of UAV imagery is that the total region covered for EPTN fault monitoring decreases. As drones fly lower to capture higher-resolution images, the total area they can cover decreases. SSD was mentioned as a less laborious and rarely used model of object identification that is capable of detecting individuals within object classes. By extracting additional detail and using it for EPTN component monitoring, more detail may be extracted and employed. Additionally, the produced objects could be utilized to estimate the location of the major EPTN component defect on the ground. Accurate categorization maps can be constructed by combining raw RGB images with supplementary products such as elevation and Structure for Motion 3D data.

Given that this project's objective is to investigate the use of drone RGB imagery for monitoring EPTN component defects, the recommended SSD model is appealing for a variety of reasons. To begin, it has been demonstrated that multiple CNN models work effectively in detecting EPTN components and single class faults. Most studies concentrate on classification methods that are applied to distinct components across multiple landscapes. A study of multi-class fault detection is offered to detect numerous EPTN component faults in a single image. SSD is one of the few one-phase models that supports object detection, which is distinct from classification in that individual items are denoted clearly. This is advantageous for monitoring EPTN component faults because it enables the generation of approximated counts.

Table 1 summarises the various features of some of the related works and their pros and cons with respect to our study. Our major contribution is that we developed a multi-class Electricity Transmission Line fault detection model. Existing literature as summarized in Table 1 were mainly focused on mono class fault detection models except Zhao et al. and Li et al. that extended theirs to two distinct faults classes. However, our model was developed and trained to detect multi-classes of faults beyond just 2 classes. Also, many of these existing models required rigorous and complex feature engineering which is made very simple with our SSD model for scalability purposes.

### Study area and dataset

The “[Study area](#)” section provides background into the study area chosen for this research. The second section (“[Datasets](#)” section) describes the electricity transmission line dataset generated for this study. Finally, the four types of electricity transmission line faults considered, are described in “[Taxonomy of faults](#)” section.

**Table 1** A summary of the pros and cons of some of the related works

Study	Data source	Dataset size	ML approach (classifier)	Detection component classes	Pros and cons per our study
Zhao et al. [30]	Cropped image from Videos from manned Helicopter	> 1200 (256 × 256)	Pre-trained AlexNet using SVM with RBF kernel. (Caffe framework)	2	Required more feature engineering their model is limited as it does not include object detection of the faults
Liu et al. [31]	UAV images	> 1500 (500 × 500)	Pre-trained faster RCNN (Caffe framework)	1	Focused only on a single type of insulators and its fault
Jiang et al. [32]	UAV aerial images (China Power Grid)	485 (300 × 300)	Ensemble learning using SSD meta-arch (TF OD API framework)	1	Again, only a single-type faults was focused on using ensemble learning model
Tao et al. [33]	UAV captured images	1956; 996 synthetic (4608 × 3456)	Pre-trained ILIN + DDN (similar mechanism with faster RCNN) (MXNet framework)	1	Single type fault and small-scale problem using a two-phase object detection (OD) method. Hence, it requires greater pre-processing of the input images
Bai et al. [35]	Picture manually from the industrial field	139 (227 × 227)	AlexNet + SPP-Net	1	A pre-trained model was utilized to transfer information and save time and computational resources. However, the model is limited to the image classification problem
Wang et al. [10]	Pictures achieved by industrial camera and mobile phone camera	510 (280 × 328)	Pre-trained CNN + object detection	1	The unique method described in this work successfully detects the power lines and its faults. However, the study explores just a unary EPTN component with
Guo et al. [37]	Helicopter captured images	250 (1024 × 1024)	RCF + FRAC	1	The dataset utilized in the model had low spatial, spectral, and geometric resolution. Consequently, the predictive model was not developed to characteristically identify broken strand faults at varying perception /distance-depth level



Table 1 (continued)

Study	Data source	Dataset size	ML approach (classifier)	Detection component classes	Pros and cons per our study
Chen et al. [38]	Images	300 (400 × 600)	SOFCN	1	This research proposes a fault detection approach employing a second-order full convolutional network (SOFCN) to overcome low-level visual feature extraction, background complexity, and classifier design methodology. This has resulted in the model complexity. A drawback to this method is the SOFCN approach requires segmented ground truth data, which are hard to annotate and limits the amount of EPTN component fault types that was considered
Liao et al. [39]	Aerial images	> 2000 (600 × 800)	Improved faster RCNN (Soft-NMS) keras	1	The proposed method achieves great result for insulator fault types and suitable for detecting insulator defect in complex backgrounds. As a two/phase OD method, major demerit to this method is that it is quite slower than the normal Faster R-CNN
Li et al. [40]	UAV images	800	ROI mining Faster RCNN	2	The paper focuses on a unique fault type—bird's nest obstruction detection. Also, the focus loss function is implemented in the RPN classification stage to balance the quantity of unbalanced data, foreground, and background data

### Study area

The study area is made up of four different transmission line corridors in Nigeria. Different transmission line corridors were explored for feasibility in this study with the help of power-line engineers and photographs from reconnaissance surveys. Six transmission line corridors were investigated in total, and four corridors were selected after reconnaissance by the ground truth team. This decision was made based on the sites' usability for field experiments and based on the spatial resolution of the acquired imagery. These transmission corridors virtually have a connection with all the 36 states in Nigeria and the Federal Capital Territory. Nigeria lies between latitudes 4° and 14° N, and longitudes 2° and 15° E. The Nigerian power transmission network called the Transmission Company of Nigeria (TCN) is responsible for the transmission of power in two phases, the 330–132 kV and the 132–33 kV through the transmission lines (otherwise referred to as conductors) [36]. In general, all transmission corridors in Nigeria share a similar structure, their infrastructure is radial and thus causes inherent problems without redundancies [37].

### Datasets

The DJI Phantom (DJI FC330) fitted with high-resolution cameras was flown across the four transmission corridors namely, Shiroro-Kaduna, Lagos, Abuja, and Enugu overhead transmission lines to capture pylons, conductors, other components of power line/pylon accessories (e.g., insulators, fittings, cross arms) as well as the surrounding features (e.g., vegetation) from varying angles. The imagery is in three spectral bands (visible RGB) with high spatial resolution. The aerial survey was conducted from October 12, 2020, to October 22, 2020. Thousands of large images tiles of the study area can be characterised as high-resolution oblique RGB images of dimension  $4000 \times 3000$  pixels (72 dpi). The mean pixel sensor resolution is 0.00124 m. Generally, within the images, the most prominent objects are located and systematically distributed transmission conductors and pylons with dirt roads, small patches of natural forest, and grasslands. We worked with the Nigerian Transmission Company for this drone mission. Also, the dataset collection was acquired in such a way that the different angle, distance, and depth adds to the distinctiveness, volume, and variety of the 'Felect' dataset. Although each inspection transmission network location has its own photographic identity, the photographs all have a comparable original pixel size of  $4000 \times 3000$  pixels. Moreover, since the natural images that contain all the transmission components' faults are scarce, we create simulated insulator missing faults images as a key step under the supervision of a power expert. The simulated transmission components' faults samples were achieved using Photoshop software since a large amount of the fault taxonomy in aerial images is scarce. We next develop a novel transmission component faults dataset, which is referred to as 'Felect' in this study, and analyze it. The acquired images were explored for viability through visual inspection. Blurred images, noisy images, and those with obstructions were discarded during the visual inspection of the images and data annotation.

### Taxonomy of faults

The main purpose of this process is to classify the faults found in the transmission components. Each transmission line component like pylons, conductors, and pylon

accessories or fittings (e.g., insulators, dampers, and fixtures), has different and unique faults.

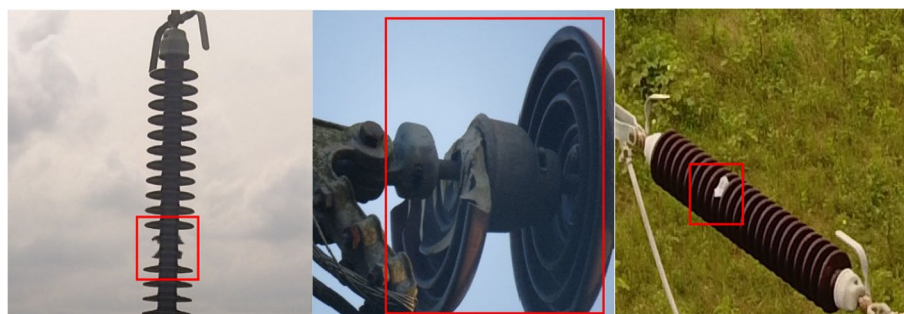
Transmission line pylons are used to extend the conductors over long distances, supporting lightning safety cables and other transmission elements. They ensure the proper electrical transmission process of the other components by preserving the original design positioning and providing sufficient grounding against adjoining objects. Insulators are critical elements in a transmission line as they protect conductors by allowing lines to retain their expected electrical insulation strength [38]. As seen in Fig. 2, the insulator has a repetitive, stacked cap structure. The colour, size, and string numbers of the insulators vary based on the transmission capacity and manufacturing design (e.g., single string and double strings). The pylon accessories, also called fittings, are the connectors of major components or elements seen in the electricity transmission lines. They mainly serve as support, inhibitors, connectors to the other transmission components. These include conductor clamps, dampers, splicing fitting, protective fittings, and guy wire fittings.

Consequently, most of these individual components have many different types of faults. For this research, the defects were divided taxonomically into four categories: missing insulator, broken insulator, rusty clamp, and broken dampers according to the contents of the captured aerial photographs. The detailed fault taxonomy discussed in this study is as follow:

- i. Missing insulator: these are glass insulators with a missing insulator cap (plate); see Fig. 1.



**Fig. 1** Missing glass insulator faults



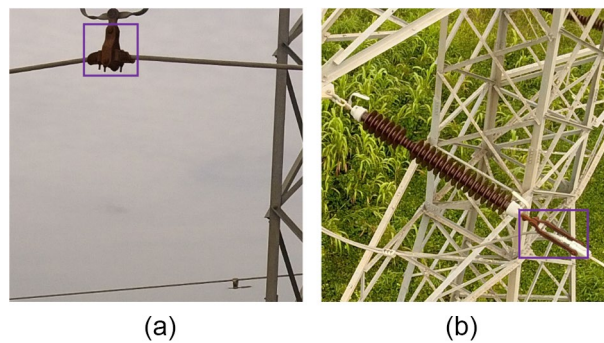
**Fig. 2** Broken insulator faults prominent with the porcelain or composite plate type insulator

- ii. Broken insulator: this applies to those insulators that are made of porcelain or composite polymer plate or cap materials. In this case, the plate is incompletely destroyed by pressure exerted by external forces such as weather, especially thunder-strike and thaw (Fig. 2).
- iii. Rusty clamp: the conductor clamp (strain or suspension clamp) helps to hold all components, especially the insulator, to the tower architecture based on its design. A faulty clamp can lead to the insulator's total malfunction, hence leading to transmission collapse (Fig. 3).
- iv. Broken fitting: broken fittings such as shown below in Fig. 4 where the vibration damper is broken could cause conductor fatigue and strand breakage.

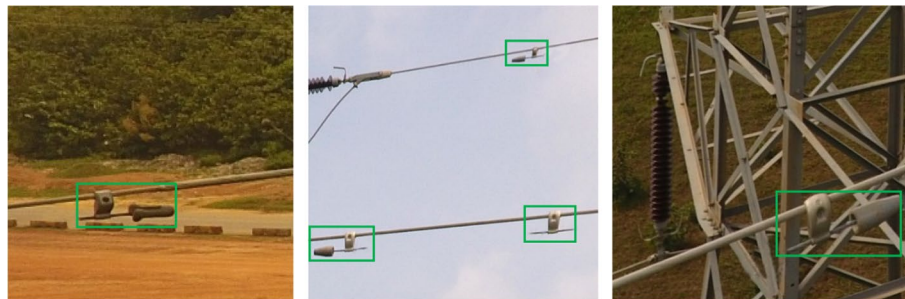
## Methodology

This section outlines the approaches and considerations for developing the predictive model for our case study transmission line fault detection from high-resolution imagery. This section also provides the algorithm description and architecture workflow of the single-shot detection models developed and designed for this study.

The proposed method is designed to detect four-class EPTN faults in a complex aerial imagery. In order to achieve a multi-level perception taking account, the small-scale problem and the depth of the convolutional neural network, the Single shot detector with FPN architecture was adapted for this purpose. The Single shot



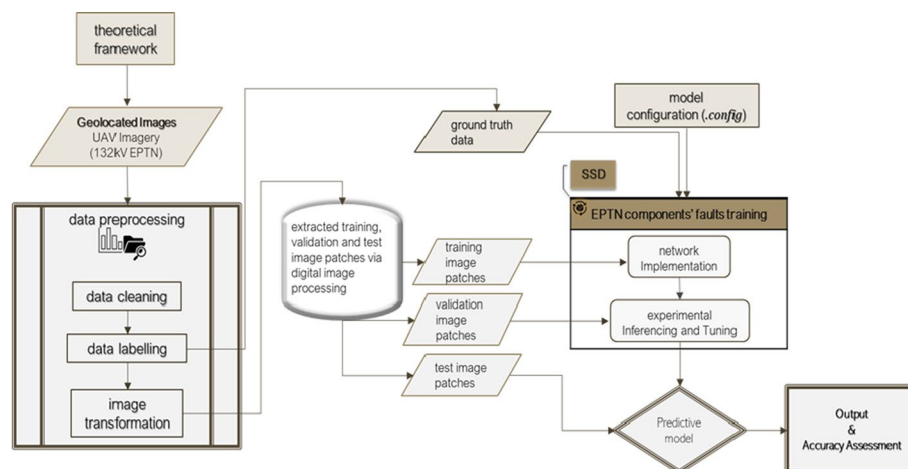
**Fig. 3** Rusty strain (a) and suspension (b) clamp



**Fig. 4** Broken fitting (vibration dampers)

detector with FPN architecture aimed to identify electric power transmission network faults based on an RGB drone imagery. The model analyzed the preprocessed images together with its corresponding annotated ground truth layer to categorize and detect the EPTN datasets into one of the four fault types: missing knob, missing insulator, broken insulator and rusty clamp. The output detection is a probable detected EPTN components' faults with an associated loss values and prediction error. The model presents a multi-scale pyramidal space network in combination with spatially informed aerial inputs to produce the detections. The development workflow and algorithm from data preparation to fault detection are described in Fig. 5 and Table 2 below.

Figure 5 describes the workflow to build the model to detect faults during EPTN inspection. First, the input RGB image which is subsequently processed are used to generate the input dataset layers. At the same time, a ground truth bounding box is created, in which the various EPTN component faults are identified in the images at pixel level. Both the preprocessed EPTN datasets and the annotated bounding box are split into training, validation, and test datasets. The SSD-FPN model uses the training dataset as its input. The model (see “[Network training](#)” section) uses the created bounding box to forecast if the current image frame contains a defective EPTN component in these terms. At the same time, utilizing higher level semantic visual representations based on contours extraction and subsequent size filtering, the bounding box and input image are utilized to extract the Region Of Interest (ROI) of every fault region in the image. This study selected three predefined backbone architectures. The three models were put to test to see which architecture best generalizes, identifies, and detects EPTN component failures, as well as to see how the spatially informed input affected the results. The training and inference output is a distribution over the model coefficients that is then engineered to determine the position of possible detections for each fault class for a particular processed image frame. The predictions are summed up to determine the final detection and performance metrics (precision, recall and F1-score). The feature pyramidal space is an approach designed to enhance the recall, multi-level perception, and



**Fig. 5** Methodology overview

**Table 2** Algorithmic workflow for the model development

---

Input: The image set, $I_{rgb}$ (current RGB image frame), $I_{bounding\ box}$ (ground truth annotation of actual components' fault types), model (SSD model).
Output: Identification of feature vectors consisting of 4 classes of all EPTN fault types.
Transmission Inspection folder: count of total number of each fault type captured from the different transmission lines' components.
1. Ground truth collection and UAV data collection
2. Data processing of $I_{rgb}$ to extract input images of size $600 \times 600$
For each image, the following step are performed
$I_{rgb} = I_{132kV} + I_{33kV}$
$I_{132kV} = F_{simulation}(I_{132kV})$
$I_{132kV} = \text{split } I_{132kV} \text{ to tiles of size } 600 * 600$
$I_{33kV} = \text{resize}_{600*600} \text{ and pad } (I_{33kV})$
Merge both $I_{132kV} + I_{33kV}$ for data labelling
Image annotation to obtain $I_{bounding\ box}$
3. Splitting the preprocessed images into train ( $I_{tr}$ ), validation ( $I_{va}$ ) and test images ( $I_{te}$ ) patches.
4. Initialization of variables: assign zero to variables i, j of the SSD network
5. while (i < transmission inspection folder) do //Read all the split TF record-oriented binary data and $I_{bounding\ box}$
6.     compute different multiscale output feature map sets from i by using the different backbone architectures.
7.     while (j < feature map) do //Read and extract each feature layer and each position in it
8.         divide j into equal parts using default grid to M x M
9.         upsample j using the factor n using the feature pyramidal scale space
10.        compute default boxes for j and <i>generate prior/anchor box using list<sub>aspect ratio</sub></i>
11.        compute positive and negative sample of EPTN fault's location
12.        compute the confidence value for <i>EPTN_fault_class</i>
13.        compute coordinates of the detected bounding boxes for the <i>EPTN_fault_class</i> pairs using NMS
14.        pick the $I_{bounding\ box}$ with the biggest probability and remove all other having $IoU > \text{threshold}$
15.        combine upsample with corresponding subsample and input images
16.        concatenate <i>EPTN_fault_class</i> and $I_{bounding\ box}$ prediction for multiple scale.
17.        move to next feature map layer for EPTN dataset record increase the value by 1
18.     end
19.     move to the next TF data representing the image in the transmission folder increase the value by 1
20. end
21. Experimental Inferencing and Tuning on $I_{va}$ and $I_{bounding\ box}$ of validation image
22. Using the Predictive model on the test image patches, $I_{te}$

---

overcome small-scale problem, which will be detailed later. The model implementation procedure is captured in an algorithmic workflow described in Table 2 below.

### Convolutional neural networks

Convolutional Neural Networks (CNNs), which are specialized neural networks developed to exploit the two-dimensional nature of images, have in recent years advanced deep learning tasks (high-level vision) such as image classification, object detection, and image segmentation, as well as low-level vision tasks such as edge detection [39]. The deep learning task (deep convNet) was first developed for image classification problems based on the performance of convolution layers to recognize edges, patterns, context, and shapes resulting in a convolution feature map having



spatial dimensions smaller and deeper than the original [40]. The progenitor of image classification architecture otherwise known as feature extractor in object detection problems is AlexNet with an 8-layer CNN, i.e., 5 convolutional layers + 3 fully connected layers developed by Krizhevsky et al. [41] in Imagenet challenge of 2012. Many improvements have been made to the architecture of Krizhevsky et al. [41] over the years. These include using a narrower receptive window and increasing the network depth.

Similarly, from the 2014 ImageNet contest, VGGNet metamorphosed intending to improve the work developed by Krizhevsky et al. [41]. This CNN architecture took first place in the localisation task and second place in the classification task. VGGNet's breakthrough is the mixture of kernel filters ( $3 \times 3$  filters) and deep neural networks (16–19 layers). The authors believed that  $3 \times 3$  convolution layers have the same efficient receptive area as the  $7 \times 7$  convolution layer, however, VGGNet's architecture is wider, with larger non-linearities, and fewer parameters to update [42]. This solidifies the concept that the best way to maximize the performance of CNNs is by increasing the depth and width of the CNNs.

The complexity of image classification problems increasingly calls for larger CNNs. However, deep CNNs with several layers can be difficult to train because of the problem of vanishing and exploding gradients. To handle this problem, the residual network learning called ResNet gained traction. Residual networks were built with shortcuts to whole networks inspired by VGG networks by the subject of skipping [43]. To dissociate with the concept of increasing depth when creating CNN architecture, ResNet proposed a shallower network using shortcut connections, i.e., directly connecting the early layer's input to a later layer. The significant ability to train very deep CNNs in 50, 101, and 152 layers with great successful connections are attributed to the regular cut-off connection (skipping) among the Deep CNN blocks [43].

The general tendency for network speeds has been to go deeper and more complex. This results in extended preparation and higher computing costs [44]. The aim of making low-latency models for mobile and embedded devices led Howard and Wang [45] to develop a lightweight deep neural network model referred to as Mobile networks (MobileNets). MobileNets and their derivatives were implemented to substitute a much deeper network constrained by the speed in achieving satisfactory output and real-time applications. This design's idea is that the regular neural network convolution layer is broken down into two filters, depth-wise convolution, and pointwise convolution [45]. The usual convolutional filter is more computationally complicated than depth-wise and pointwise convolutions. To achieve this model implementation, each channel is convolved with its kernel, called a depthwise convolution. Next, the pointwise ( $1 \times 1$ ) convolution is processed to abstract and integrate the individual intermediate output from the depth-wise convolution into a single feature layer.

Inspired by the success of CNNs in image classification and the need to adapt CNNs for more complex tasks other than classification problems, the object detection approach was conceived, which comprises the classification of objects and finding objects of interest positions in the image via regression. In line with this thought, the Faster R-CNN was developed utilizing a region-based CNN. Faster R-CNN performs object detection using two major modules: a Regional Proposal Network (RPN) proposing regions,

and a Region-CNN (R-CNN) detector classifying regions and refining boundary boxes. The model involves first the use of a base network, i.e., CNN architecture pretrained for classification to generate the necessary activation feature map [46]. Then, the extracted feature maps are passed through the RPN to generate an object proposal. Each object proposal from the RPN, is then applied in the network by overlapping them over the existing convolutional feature map. This extracts various fixed feature maps of the field of interest for each proposal. The final Region-based CNNs (R-CNN) incorporate the preceding output with class details based on the region's proposal. Using the object proposals extracted via RPN and the extracted features for any one of the proposals (via ROI pooling), a final class and object localisation is achieved [46]. R-CNN is a model which attempts to simulate the final phases of CNN classification where a flattened layer is applied to generate a score for each conceivable object form [34]. R-CNN has two separate objectives: classify the proposal and modify the bounding box for the proposal according to the predicted class. Although faster R-CNN is extremely reliable, it is very slow.

In the same vein, the Region-based Fully Convolutional Network (R-FCN) was developed by Dai et al. [47] to tackle the shortcomings of the initially designed Faster R-CNN frameworks. Instead of using an inefficient sub-network for each region hundreds of times, R-FCN adopts an entirely convolutional architecture over the whole image. In a way that allows network convolutions to carry out one calculation in detail and accurately, the R-FCN provides new location-sensitive scoring maps. Also, the issue between translation invariance and translation difference in recognising objects is addressed more effectively. Therefore, R-FCN integrates feature maps and applies convolution to construct position-sensitive score maps, which enable convolutional networks to successfully perform both classification and detection in a single evaluation. The position-sensitive ROI pooling is used to produce a vote array of the output size for any ROI to achieve a 2D score map of each class. For regression of the boundary box, another convolution filter is used to construct a 3D output map on the final feature maps. Then, the ROI-pooling is used to measure a 2D array with each element that includes a boundary. The sum of these elements is the final bounding box estimate [47]. RFCN presents new position-invariant spatial scores which enable convolutional networks to successfully perform both classification and detection in a single evaluation. R-FCN incorporating these enhancements allows the framework to run faster about 2 to 20 times the speed and have better accuracy; therefore, the frameworks are quick and precise but have complicated pipelines.

To aid in real-time object detection maintaining a balance between time, speed, and accuracy, many single-phase deep learning-based approaches, which detect multiple objects in a single scan, have been proposed. The two most popular single-shot models are the 'You Only Look Once' (YOLO) and Single-shot detector (SSD) frameworks. YOLO is a network that classifies bounding boxes in real-time [48]. To fulfil this, YOLO combines area proposal and region classification to form a single network and does this as the frame is simply regressing on box localization and related probabilities. YOLO uses a grid that separates the input image. The grids evaluate the bounding box position, assign confidence ratings, and conditional class probabilities. YOLO is incredibly fast because it is single-threaded; however, YOLO lacks the precision seen in the two-phase

frameworks such as R-FCN and Faster R-CNN previously discussed. The SSD is a better approach as it is focused on a feed-forward-based convolution network generating a fixed-size bounding box set and scores of object instances present in these boxes, and a final detection process based on a Non-maximum Suppression (NMS) criterion [49]. The early network layers are constructed on a standard image-classification architecture known as the base network (i.e., the classification layer without the flattened fully connected layer).

SSD supersedes its counterpart, YOLO, by introducing several modifications: (i) multi-feature maps from subsequent networking stage are predicted to allow multiscale detection; (ii) object classes and offsets at bounding box locations are predicted using regular sized small convolutional filter; and (iii) after deriving final feature map, different predictors (classifiers) are used to identify objects at varying aspect ratios in the form of feature pyramids [50]. SSDs comprise two main parts: a feature map extractor and the convolution filter for object detection. SSD attaches additional convolutional layers (feature layers), i.e., multiscale features and default boxes, which causes a steady decrease in size up to the end of the primary network [49]. Hence, the predictions of detected objects are produced at multiple levels. Unlike YOLO, which uses a fully connected layer to make predictions, the SSD adds a series of small convolutional filters to each added feature layer (or an existing one in the base network optionally) and uses them in boundary box positions to predict classes and offsets of objects [51]. These changes improve both the speed and the accuracy of SSD over YOLO.

Undoubtedly, convolutional network tasks typically have a significant role in image classification and object identification. One of how CNN achieves this high performance is via the gradient-based learning process, more specifically loss computation and the loss function [39, 52, 53]. This is believed to be the object's real value, versus the expected value. For instance, if the expected value ends up being 0.75, and the actual value is 1, the loss would be 0.25. As iterations continue, the model will better approximate the object's true value. In this respect, the optimisation process is employed so that the prediction capacity can be maximized. Mathematically, this implies that for neural networks, the loss is normally the sum of negative log probability and residual sum of squares for the classification and regression part, respectively [54, 55]. After that, the key goal is to mitigate the loss with respect to model parameters by modifying the weight vector values using neural networks. For all object detection models, the loss function is a combination of the localization (bounding box regression) and the confidence loss (object classification).

### **Data pre-processing and labelling**

Most of these individual components have many different types of faults. As this study aims to identify common electrical faults in relation to common transmission components, the dominant transmission components' faults were established to be (1) missing insulator, (2) broken insulator, (3) rusty clamp, and (4) broken fittings. Although this project undertakes to detect these four transmission components' faults, the ability of the model to also classify commonly occurring faults was investigated.

Ground truth collection consisted of field visits to the sites and convenient sampling where there are no forest trees touching transmission networks were chosen and areas

with obstruction were excluded. Identifying the most common transmission components' faults was determined and later analysed by power technicians and electrical engineers working in the Transmission Company of Nigeria (TCN), who assisted to perform a visual inspection of the aerial images to determine powerline components faults from drone imagery. The labelling tool is used to label the locations of transmission component faults, scanning through thousands of image tiles in order to establish the ground truth of all faults recognized by the power specialist.

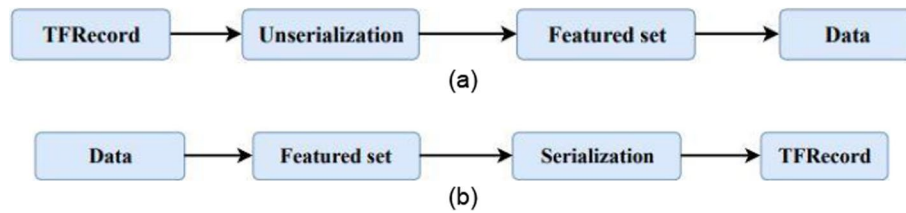
Further pre-processing of the dataset entailed a series of steps aimed at cleaning and standardizing the raw data prior to modelling. Pre-processing is critical for increasing the sensitivity of the model and validating any model that uses aerial imagery for transmission line fault detection. The entire dataset is made up of 294 images. Due to the small-scale problem identified in some research [56], the dataset (132 kV) was split into about 817 tiles centred on at least one components' fault of interest. For the other dataset representing the other 33 kV transmission line, the non-destructive resize, i.e., resize and pad approach, is applied to preserve the image aspect ratio to preserve the geometric and spatial information. Moreover, the split and resized RGB images were normalized to the same size of  $600 \times 600$  pixels following Huang et al. [50], combined to form a total of 1027 'Felect' dataset sample imagery. The data is divided into train, test, and validation sets. It was assured that 17% of the original dataset was allocated for the test dataset, and 83% of the dataset was reserved for training and validation. About 80% of training was used as the training samples, while the remaining 20% was dedicated to validation samples 2 displays the data slicing information. The drone captured the 'Felect' dataset with numerous characteristics, including diverse perspectives, sizes, occlusion, background clutter, and intra-class variance.

Thus, a "stratified" data division is used, making the proportion of the faulty components for the dataset similar to the number of images, as well as the average number of components and the intraclass variation shared equally for samples with different types of difficulties to be learnt and appropriately located and classified.

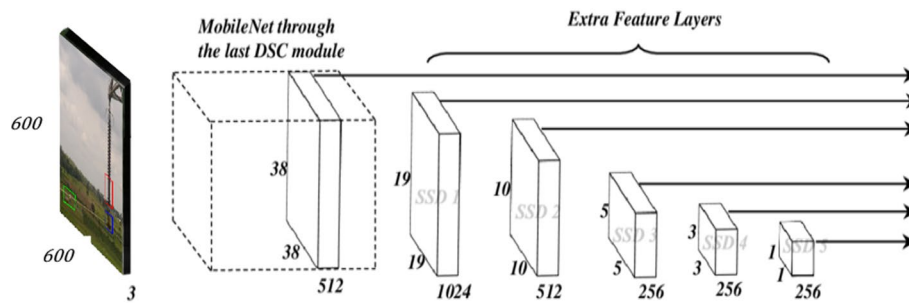
Data annotation was carried out to identify and label the training dataset for model training. The bounding box approach and pixel-wise object segmentation are two approaches that can be used to annotate the main object on the image manually [55]. To annotate the faults, the ground truth annotation of actual components' fault types was generated as a rectangular bounding box was used. A tool called 'LabelImg' was used to label the different component faults as shown in the taxonomy of faults. The details of the image, bounding box, and object class, along with shared characteristics, were stored as a VOC2007/extensible mark-up language (.xml) file. After annotating all the frames, the whole split dataset containing image patches tensor and their output label were converted into a TF record-oriented binary as depicted in Fig. 5 to help dataset initialization and ease network architecture using the TFRecordWriter function.

### Network training

As stated in "Data pre-processing and labelling" section, input patch images are first translated to tensors (TF records) with a  $[600 \times 600 \times 3]$  form before feeding it into the backbone architecture and are distributed by the action of the convolution layer to an intermediate layer consisting of a convolutional activation map. The head of the network



**Fig. 6** TRecord (a) reading and (b) writing principle



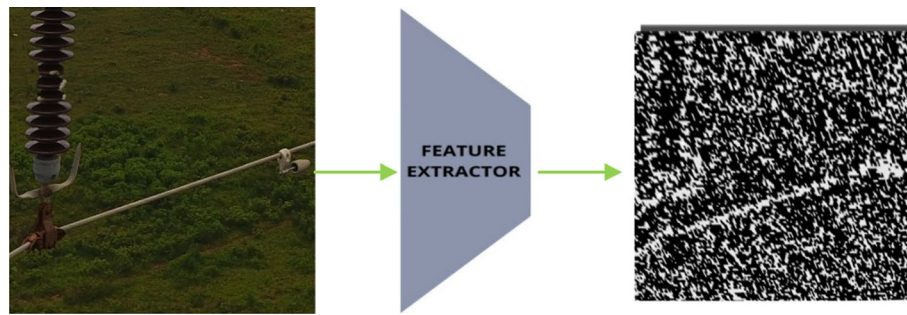
**Fig. 7** Model architecture [59]

architecture (backbone network) typically follows the patch-based CNN architecture. Therefore, image patches that contain either a single class of faults or a combination of different components' faults centred in the pixel of interest, also termed valid patches, were extracted. For, backbone neural network ResNet50, MobileNet, or ResNet101 are utilized for the first part of the SSD network as the head to develop three models.

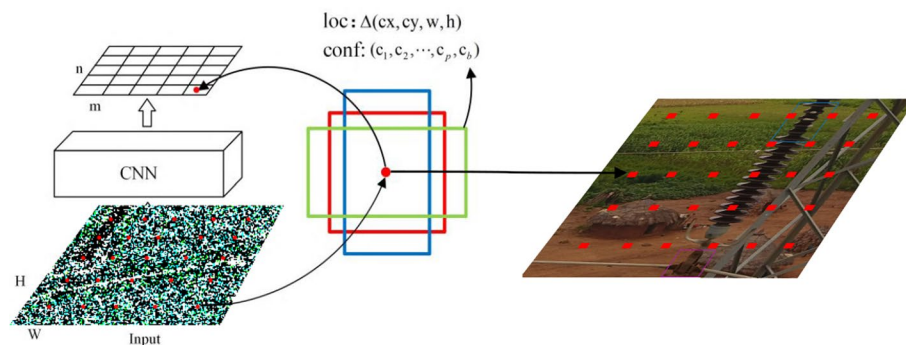
This head is made of CNN that detects smaller characteristics (patterns and corners), and later layers detect higher characteristics successively. The image was resized first into  $640 \text{ px} \times 640 \text{ px} \times 3$  (RGB) and then translated into a  $38 \times 38 \times 512$  characteristic mapping through the backbone network passed to the Conv7 denoted as SSD 1 (auxiliary layer) in Fig. 6. In all experimentation cases, the input patch tensor was abstracted into multi-level representations to classify the different faults after going through the backbone architecture (Fig. 7: without a fully connected layer). As a deep neural network, the backbone algorithm derives semantic significance from the image while maintaining its spatial structure.

The series of auxiliary convolutional layers (SSD layers) introduced after the SSD model's backbone allows the extraction of features at different scales as the input feature map decreases at each successive layer. This ensures the certainty of boundary variance and class prediction of targets at various scales. For each decreasing successive auxiliary layer (multi-scale feature maps), SSD networks grids the image and assign each grid with the task of detecting objects (Fig. 8). After this,  $3 \times 3$  convolution filters are applied to each cell to make predictions. If no object appears, the context class is not considered, and the location is ignored. Each cell in the grid will decide the location and shape of the object inside it.





**Fig. 8** Input image patch and corresponding feature map generated by the feature extractor (backbone architecture)



**Fig. 9** The default boxes generation for one cell over the backbone network feature map

Immediately after gridding the auxiliary layer, i.e., feature map at multi-level, default boxes are generated at each grid cell for each convolution layer level using a defined scale value (Fig. 9).

This scale increases progressively towards the least spatial resolution feature map level (SSD 5). Next, bounding boxes are generated via a process called default box generation (prior). Default boundary boxes generated by feature maps are selected explicitly, which are pre-computed, fixed-size boxes that closely fit the ground truth boxes. The size of the default bounding boxes depends on the input size ( $W, H$ ), scale  $s_k$  of the  $k$ th layer, and aspect ratio,  $a_r \in \{\geq 1\}$ . With the different experiment scale value,  $s_k$ , and the aspect ratio,  $a_r \in \{1.0, 2.0, 0.5\}$ , the default boxes sizes are built. The size of default boxes ( $W_d, H_d$ ) can be computed as:

$$W_d = W s_k \sqrt{a_r}, \quad H_d = H s_k \sqrt{a_r}.$$

To detect larger objects, SSD uses lower resolution layers such as the SSD 4 and SSD 5 layers in Fig. 5. Each grid prediction composition includes a boundary box defined by  $c_x, c_y, w, h$ , and four scores for each class, i.e., components faults, in the prediction, with the highest-class score associated with the positioned default bounding box. The class score,  $(c_1, \llbracket c \rrbracket_2, c_3, c_4, c_{\text{background}})$  corresponds to object classification labelled in this research as “broken insulator,” “missing insulator,” “missing knob,” and “rusty clamp.” Having these several forecasts at once and



awarding class scores to each is referred to as the Multibox. There are four predictions for every cell, regardless of the feature map's spatial resolution, and an extra one prediction to represent objectless.

To improve the SSD to detect small-scale faults type, the feature pyramid network (FPN) training structure is used in conjunction with the most immediate output feature map activated from the base network architecture. This method also imbues low-level CNN layers with more assertive semantic representation, such as layers near its head to detect small-scale object labels. In particular, the default boxes are chosen so that their Intersection over Union (IoU) is greater than 0.6.

The Sigmoid function is then performed on the output feature map generated by the last CNN to obtain a class prediction score. Thereafter, the total loss is achieved by combining the two losses obtained for backpropagation. The two new losses measured by the network for each bounding box include:

- a. The localisation loss is achieved using the weighted smooth-L<sub>1</sub> loss, calculated by comparing the generated default boxes (prior) against GT labels.

$$L_{loc}(x, l, g) = \sum_{i \in Pos} \sum_{m \in Box} x_{ij}^k smooth_{L1} \left( l_i^m - \hat{g}_j^m \right).$$

$$\hat{g}_j^{cx} = (g_j^{cx} - d_i^{cx}) / d_i^w, \hat{g}_j^{cy} = (g_j^{cy} - d_i^{cy}) / d_i^h,$$

$$\hat{g}_j^w = \log \left( \frac{g_j^w}{d_i^w} \right), \hat{g}_j^h = \log \left( \frac{g_j^h}{d_i^h} \right),$$

where  $l$  refers to the predicted box,  $g$  meaning the ground-truth box, and  $d$  refers to the default box, the 4 shape offsets  $m \in \{cx, cy, w, h\}$  are defined as the center  $(cx, cy)$  of the bounding box and its width ( $w$ ) and height ( $h$ ). Note that the predicted box and the default box are corresponding one by one. The SmoothL1 is denoted as:

$$Smooth_{L1}(X) = \begin{cases} 0.5(X)^2 & \text{if } |X| < 1 \\ |X| - 0.5 & \text{otherwise} \end{cases},$$

$$\text{where } X = l_i^m - \hat{g}_j^m.$$

- b. The confidence loss is achieved using a similar method applied in image classification, in this case, the weighted sigmoid focal.

$$L_{conf} = FL(p_t) = -\alpha_t(1 - p_t)^\gamma \log(p_t),$$

$$p_t = \begin{cases} p & \text{if } y = 1 \\ 1 - p & \text{otherwise} \end{cases},$$

where  $\gamma$  is 2.0 and  $\alpha$  equals 0.25.  $p \in [0, 1]$  is the model's estimated probability for the class with label  $y \in [0, 1]$ .  $(1 - p_t)^\gamma$  is the modulating factor for the cross-entropy loss.  $\alpha$  is the balanced variant of the focal loss.

The default boxes that did not get scored against any ground truth boxes are viewed as negatively matchboxes and are applied to only the confidence loss, while the positive box is applied to the overall loss. This loss value is back propagated to update the network parameters using different optimizers during experimentation.

### The feature pyramid network

Feature pyramid network (FPN) integrates strong semantics with weak semantics, i.e., it takes the single-scale aerial images as inputs, generating proportionally sized feature maps at multiple levels in a fully convolutional paradigm [57]. In this case, from the initial output feature maps the architecture consists of two pathways: Bottom-up and Top-down pathways. The input images, i.e., the output multi-scale feature maps (conv.) derived from several layers as inputs first go through the bottom-up pathway (using  $1 \times 1$  convolutions) and produce a feature map at each stage. The outputs from the convolutional network from the bottom-up pathways are combined with convolutional layers to produce inputs, which are then used for the top-down pathway. The convolution from the bottom-up and that of the top-down are combined using a lateral link, they have similar filter sizes/channels in their feature maps. This finally helps us merge low-resolution features with high-resolution features so that we can upsample the feature of improved resolution.

We naturally depict the higher-level employing lower-level pixel visual attributes. In the convolution layer, the revised feature maps are concatenated with the original maps and scaled to the accurate filters. Higher-resolution features are upsampled from coarser but semantically more robust feature maps. The spatial resolution is doubled during upsampling, and the nearest neighbour is utilized for simplicity. The bottom-up is constructed using two convolutional blocks consisting of 3 convolutional units (3 alternating convolutional and pooling layers in 3 units). The top-down pathway is composed of six alternating layers of convolutional and pooling blocks, three of which are for projection and the remaining three for smoothing the combined lateral link and top-down path to create the final feature map to mitigate the aliasing effect of upsampling. Finally, the outputs of the two chunks are concatenated and fed to adjacent fully connected layers. The output of the last fully connected layer holds the box predictor and class predictor and is compared to the associated labels to calculate the performance metrics.

### Experimental design

- i. The current projects' fundamental problems were related to the number of computing resources required and the dataset's limited size. In this study, the experiments—backbone architecture and meta-architecture were built on top of the deep learning framework of TensorFlow Object Detection API (TF 1) Model Zoo. Two separate outlets were utilized for execution, they include A physical computer with AMD Ryzen 5 3550H with Radeon Vega Mobile Rfx processor CPU with 7.81 GB for data processing, preparation, and model testing.

**Table 3** Data partition

Dataset	#Components' faults	%Components' faults	#Images	%Images	Missing knob	Broken ins.	Missing ins.	Rusty clamp
Train	1198	62.7	646	62.9	490	223	259	225
Validation	372	19.5	207	20	151	70	82	68
Test	340	17.8	154	16.8	142	75	49	73
Total	1910	100	1007	99.7	783	368	390	366

**Table 4** Training hyperparameters settings for CNN models

Hyperparameters	Values
Momentum	0.9
L <sub>2</sub> regularization	0.0004
Batch size	8
IoU-threshold	0.6
Min and max scale	3–7

- ii. Google Colab environment on the Google cloud server with 2 Intel(R) Xeon(R) @ 2.20 GHz processor CPU with 13 GB RAM (200 GB free space disk) and 1 GPU (Tesla K80) with 12.6 GB RAM for parallel processing for experimentation.

To ensure optimal experimentation with the data available, the validation dataset was utilized for evaluating the trained network. Due to computation cost and speed, the k fold cross-validation was not implemented. Hence, a hold-out validation with shuffling was used to generate an average detection result for all the models.

The training and test sets were used for the network training and testing, while the validation set as described in Table 3 below, was used to tune the hyperparameters. In the NMS process, 100 detections and an IoU threshold of 0.6 were maintained for each class. The momentum and the batch size were set as 0.9 and 8. The regularization value was set to 0.0004 as shown in Table 4. The warm-up learning rate of 0.0001333 was used to assist in the weight optimization after 5000 training steps and at the end of the training, the period decayed to zero. Batch normalization (BN) [60, 61] is used after the convolution layer and before nonlinearity layers to avoid overfitting and to save time during hyperparameter tuning [62]. During training, the data augmentation technique was used to increase samples' diversity because of insufficient training data. Six methods were employed for this data augmentation in the training phase: jitter boxes, horizontal flip, vertical flip, crop, pixel value, and rotation. To ensure guaranteed detection, the IOU confidence level is set at 0.6. Five measurements, including recall, precision,  $f_1$  score, average accuracy, and mAP, are applied to evaluate the components' faults model performance.

## Result and discussion

According to previous research, a popular approach that is gaining popularity for EPTN fault inspection and maintenance via remote sensing is the application of Deep Learning. Most deep learning methods utilize the two-stage object detection architecture and are

utilized for unary classification. While data augmentation has been used to alleviate the data deficit, pre-trained models and unsupervised learning have also been tested. To pick the suitable methodological components in this investigation, we considered the type of input imagery (oblique optical imagery) and the intended number of detection (four EPTN component faults). According to the literature review, we addressed the data shortage in two unique ways: (1) enriching the dataset to make it acceptable for training a Deep Architecture, and (2) transfer learning with a benchmark dataset. Three distinct models were then developed and compared. Through the introduction of FPN, faults that are predominantly seen as minute in EPTN components and captured as small aspect ratios on images with low perception were appropriately recognized in the same way as those found in large aspect ratios. To start with, a train, validation, and test set from the 'Felect' dataset were utilized to determine the time required to run on a smaller dataset. The default hyperparameters were employed, with the learning rate, backbone, and pretrained weights from benchmark datasets being the most critical. The model can be trained completely from scratch, updating the weights for all layers in the process. However, other comparable studies discovered that by freezing the training layers and using pre-trained MS Coco or ImageNet weights, which are large-scale image datasets, the model's ability to learn and detect objects improved significantly [41, 43]. As a result, the term "transfer learning" was coined and widely adopted. Transfer learning is the process of transferring knowledge from a previously trained model to a new dataset [20, 31]. This also helps the model run faster, as it makes inferences using weights from previously learnt objects. Both of these benchmark datasets enable the model to start learning from an established machine learning checkpoint rather than from the beginning. Initial training with training images took about 50 h to complete on a local CPU. This was found to be excessively time-consuming, and a more suitable strategy for training the model was investigated.

The three different pretrained models (backbone architecture) were trained with the same parameters and the same training, validation, and testing datasets and improved using hyperparameters tuning (see "[Data pre-processing and labelling](#)" section). After running hyperparameter refinement simulations, the most optimum value was recorded and incorporated into each model to achieve the localisation and classification of the different EPTN component faults. Using these proposed SSD models with different backbones called SSD MobNet, SSD Rest101, and SSD Rest50, a four-class EPTN fault object detection was performed on our testing dataset containing 142 missing knobs, 75 broken insulators, 73 rusty clamps, and 45 missing insulator plate faults. The models were tested using three separate metrics, including F1-score and mAP. As previously mentioned in "[The feature pyramid network](#)" section, a holdout validation scheme was employed to produce an average detection result for all the utilized models in the study area. As observed, the CNN-based networks tested perform considerably well (regardless of the experimental setting considered), indicating CNN's superior capability to accurately detect faults on transmission assets in Nigeria using drone imagery. On the one hand, CNNs' remarkable ability to extract incredibly feature vectors from a neighbouring region enables the generation of more precise detections for a given pixel. On the other hand, the spatial resolution of drone imagery (in comparison to other conventional -space-borne sensors, for example, Landsat and sentinel) may make these convolutional features even more informative for identifying and diagnosing faults in the

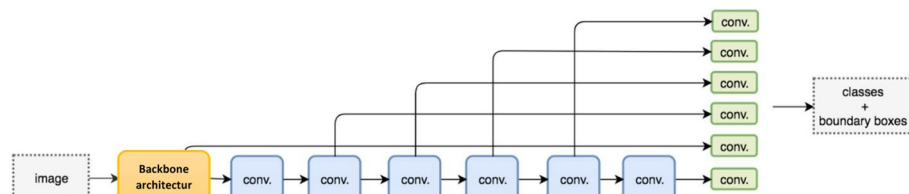
**Table 5** Assessment of SSD Rest101, SSD Rest50, and SSD MobNet on the test dataset

Models	ETPN classes	Precision	Recall	F1-score	Avg F1	mAP	Models
SSD Rest101	Broken insulator	98.33	78.67	87.41	81.93	88.7	SSD Rest101
	Missing insulator	100	67.34	80.49			SSD Rest101
	Missing knob	96.51	58.45	72.8			SSD Rest101
	Rusty clamp	98.27	78.08	87.02			SSD Rest101
SSD Rest50	Broken insulator	100	72	83.72	82.54	89.61	SSD Rest50
	Missing insulator	100	57.14	72.73			SSD Rest50
	Missing knob	97.22	73.94	84			SSD Rest50
	Rusty clamp	96.82	83.56	89.71			SSD Rest50
SSD MobNet	Broken insulator	100	68	80.95	76.8825	82.98	SSD MobNet
	Missing insulator	95.83	46.93	63.01			SSD MobNet
	Missing knob	95.83	65.24	77.64			SSD MobNet
	Rusty clamp	93.55	79.45	85.93			SSD MobNet

context of this work. This finding is consistent with previous research utilizing the SSD meta-architecture for model training and evaluation. Jiang et al. [32] examined various deep meta-architectures with their proposed ensemble model produced using SSD (one-stage object detection model), concluding that the latter is more effective in monitoring the condition of insulators via the detection network.

In this work, we have utilized three different optimizers (RMSprop, momentum, Adam) and the best average results were always achieved with the momentum optimizer. In general, it was observed that the momentum optimizer gave the best mAP across the different models using the default hyperparameter settings. SSD Rest50, SSD Rest101, and SSD MobNet achieved an mAP of 82.85%, 80.42%, 79.61%, respectively, using the momentum optimizer. The SSD Rest50 gained the highest accuracy when compared to the other two models. Also, the experiment showed that for the momentum optimizer, the validation, and total loss converge optimally. Furthermore, it has been expressively proven that the model's convergence is affected by the optimizer utilized. We observed that all the optimizers attain acceptable rates of accuracy, but one of the most glaring differences is the value of training loss and validation loss as well as the model convergence, i.e., the degree of loss range from zero. It can be inferred that the optimizer momentum with cosine learning rate is the one that provides the best results and the quickest to converge.

Using Momentum as the ideal learning algorithm, numerous learning rate settings were checked to improve the model performance. After several preliminary evaluations, it was confirmed that the best initial Learning rate (Lr) was 0.09. The first model, SSD MobNet, reached an mAP of 73.94%, 71.56%, 79.61%, and 82.52%, with the learning rate was 0.001, 0.01, 0.05, and 0.09, respectively, better performance of the model with increasing learning rate value. Similarly, the remaining two models: SSD Rest50 and SSD Rest101, demonstrated the greatest average mAP of 86.29% and 83.14%, with a learning rate of 0.09, which is 3.44% and 2.72% higher than those obtained when set to 0.05. The learning rate plays a significant role in the network's performance and how easily it can generalize [58]. Specifically, decreasing the learning rate beneath this value (0.09), which gives the fastest convergence outcomes, will improve the mAP to generalize, particularly



**Fig. 10** Multiscale downscaling layer (auxiliary layer) concept

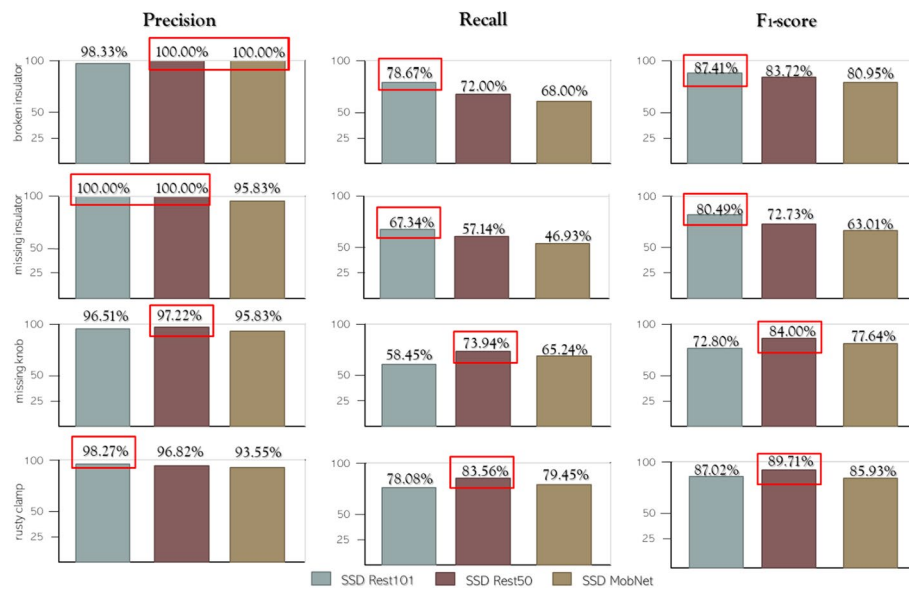
for large, dynamic cases. The learning rate used for all models was 0.09 as they all performed better with this value.

The test results of the proposed single-phase components' faults identification and classification pipeline are shown in Table 5. It illustrates the precision, recall, f1 score, and accuracy of the three models, respectively. As can be seen from Table 5, the SSD ResNet holds the maximum overall mAP score of about 89.61% for the components' faults detected and properly classified. Low precision rates suggest that a significant number of false-positive samples of the different EPTN component faults are generated when using the models for fault classification [22, 32], which is not the case here as the model generated fewer false-positive samples of EPTN faults; hence the reason for the general precision rate being above 90.9%. By delving deeper into these results, we can develop a greater understanding of the contextual factors between the models and the various exploratory scenarios considered.

With regards to the research studies under consideration, there are more component faults not identified than misclassified, causing a lower recall rate, especially for the missing knob fault type as shown in Fig. 10. From Fig. 10, the recall rate of the SSD Rest50 is 57.14%, 73.94%, and 83.56% for missing knob and rusty clamp fault classes, respectively, which varies from about 15.50%, and 5.48% to that detected and classified by SSD Rest101. Alternatively, the recall rate for the SSD Rest 101 is the greatest in identifying the broken and missing insulator faults. The SSD Rest50 achieved a better recall rate for broken insulator cap, missing insulator cap, missing knob, and rusty clamp component fault classes compared to SSD MobNet by 4.00%, 10.21%, 8.70%, and 4.11%. The SSD MobNet performs the least in detecting and classifying the missing insulator fault class compared to that the SSD Rest101 and SSD Rest50 models. Generally, all models had a satisfactory recall in detecting and classifying each fault class, especially when identifying missing knob and rusty clamp faults. This reveals that the experimental single-stage components' fault detection and classification pipeline can solve this identified problem by substantially increasing the model's performance in identifying and classifying the EPTN faults.

The SSD Rest101 is the second-best model with an overall mAP of 88.70%. Of the object detection methods tested, the one that delivered the least prediction (82.98%) was SSD MobNet. While the ResNet 101 derived model termed the SSD Rest101 has been noted to be the best in principle [33, 43]; however, in this case, the SSD model based on ResNet 50 contrasts conventional assumptions by revealing an improved result. The complexity of the network architecture can indeed justify the explanation behind the persistent lower results by the ResNet 101 model, which is made of much deeper layers in contrast to the size of the training dataset; making the model





**Fig. 11** A comparative view of the performance metrics

characteristics over subsample and learns features; thus, affecting the performance in detecting different components' faults optimally. Furthermore, to intuitively reflect the proposed model's detection performance, the loss value graph was evaluated to understand, rationalize, and justify the proposed models' generalization ability. In general, we can assess the proposed model's performance using the loss graphs and examine the group of classification, localisation, and regularization loss [53, 58]. Figure 11 gives snapshots of the loss value sensitivity over the training and validation phase through the network trajectory.

A good performance is established based on the total and validation loss decrease until it becomes stable and the difference between both loss values reaches a minimum [58]. If the prediction errors are unbiased, the validation error should be near zero, and the validation loss decreases with a decrease in training loss. This can be seen distinctively by the loss graph of SSD Rest50, SSD Rest101, and SSD MobNet model. The Rest50 model represents a Deep network, the SSD Rest 101 serves as a super Deep Network, while the SSD MobNet is a shallow network.

The various weight optimizations associated with the training and validation of the dataset based on the model architecture show that the loss value remained relatively stable. In the experiments, the base and top CNN layers used the Rectified Linear Units (ReLUs) as activation functions over shuffled mini-batch gradient descent (batch size of 8) with the Adam optimization algorithm. The final output uses a sigmoid function for each decision node. Using the sigmoid activation, the final achieved pair losses, i.e., [validation loss, training loss] for the SSD MobNet, SSD Rest50, and SSD Rest101, were approximately equal to [0.281, 0.309], [0.378, 0.385] and [0.356, 0.342] respectively. In contrast to the SSD MobNet, SSD Rest50 and SSD Rest101 have higher orders of magnitude as they have more parameters due to having more layers and more filters per layer. This allowed the model to learn more complex



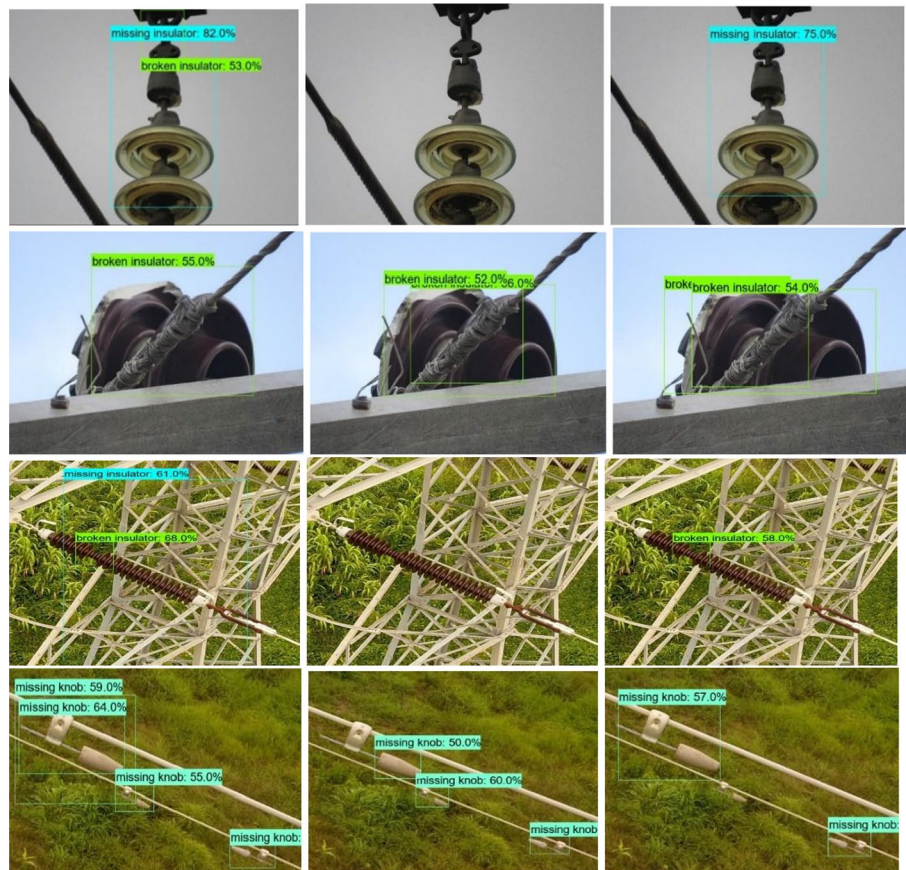
**Fig. 12** Epoch vs. loss graphs over time

features than the shallow network can provide. In the SSD Rest101, it is observable that the dataset was not sufficient to train the deeper network. The ResNet 50 backbone architecture, which represents the Deep Network, performs much better in minimizing the loss values than either previous network, achieving train and validation losses of 0.378 and 0.385, respectively, after 15 epochs. To better understand the algorithms proposed, some of the networks' training and development images output were examined. Finally, there is a strong link between training loss and validation loss. They both decrease and then become stable at a constant value. This suggests that the model is correctly trained and has a high probability of working well on any dataset within this use case.

Overall, the proposed network consistently performs well in all tested scenarios, indicating that it is suitable for detecting faults on power lines in Nigeria using UAV imagery.

The primary advantage of the proposed architecture over alternative methods is the SSD characteristic, which is based on its ability to effectively utilize a single-phase method for fault diagnosis of electricity transmission tasks and on its ability to effectively balance contextual constraints. Figure 12 provides an example of all the output images produced by all the models implemented. The sky-blue box denotes the missing insulator; the green box denotes the broken insulator, the turquoise box denotes the missing knob faults, while the white box bounds the rusty clamp defects. Each box is marked by the components' faults and its confidence score. The first column to the third column depicts the implemented method's performance, SSD MobNet, SSD Rest101, and SSD Rest50, respectively (Fig. 13).

In the first row, the SSD MobNet (leftmost) gives an accurate detection of the missing insulator plate with a false positive identification of a broken insulator, SSD Rest101 (middle) gives no result even with the presence of a missing insulator plate and the SSD Rest50 (rightmost) achieves the best result with no false prediction. In the second row of Fig. 6, the SSD Rest50 method detects the broken insulator fault, while the other implemented model leads to a wrong judgment with a false rate. In the third row, the model's performance behaves similarly to what is observed in the first row as the model is affected by the convoluted background interferences. The fourth row shows that all the implemented models had depicted the missing knob near perfectly with just one false



**Fig. 13** Experimental results of the four components' faults. The first column to the third column depicts the proposed method's performance in each row, SSD MobNet, SSD Rest50, and SSD Rest101

positive of the missing knob faults for SSD MobNet (leftmost) and one false negative (rightmost).

### Conclusion and future work

This study has expounded the incredible potential of combining UAV surveillance imagery and deep learning for automatic power transmission line inspection and fault detection, especially in developing countries. To approximate a real-world situation in which available RGB imageries are scarce, if not non-existent, the various explored possibilities address data scarcity and imbalance through the use of transfer learning strategies, paving the way for a novel approach to the difficult problem of multi-class EPTN fault identification based on limited data. The experimentation design of this study validates our proposed utilization of the deep learning model on UAV imagery for power line fault detection. A comparative analysis of different state-of-the-art manual and deep learning-based power line fault detection techniques was carried out.

The findings of this study allow for the drawing of several significant conclusions about the general use of deep learning and UAV imagery for this application. First, transfer learning provided a better strategy to achieve a robust performance for all fault classes, being able to predict correctly more than half of their instances. Also, the adaptive optimizer, momentum with mini-batch SGD, allowed for the faster convergence of the proposed model and automatically predict the optimum learning rate. Second, it was observed that a higher learning rate achieved better mAP values across all the models implemented. When examined more closely, each of the three approaches has a unique effect on each class, with SSDRest50 achieving the best performance. One could argue that training a multi-class detection model on such a massive image dataset and the egregiously skewed dataset is difficult and a 'big data' problem. Based on the practical insight derived from comparing the detections of the models to the ground truth, we may assume that additional work is required to build a generalized classifier utilizing the SSD FPN meta-architecture that is faster and with higher precision and recall values.

With these considerations in mind, the presented modelling approach addresses the challenges of using accessible UAV imagery in conjunction with data from developing countries to automate the monitoring of electrical power transmission faults in the future, thereby contributing to more reliable and formative transmission companies and power industry practices. In the future, the single-stage component identification and classification pipeline should be expanded to account for faults in different components' shapes and severity levels. Also, to measure the magnitude of the detected defects' scale, applying instance segmentation and using this knowledge to measure the scale and magnitude of the faults might suffice. In the future, given the limited data available for the fault inspection process, there are two methods to solve this problem. These include foreground and background superposition using segmentation networks and image processing techniques, and Generative Adversarial Networks (GANs) to create synthetic images. Additionally, extending this work to cover real-time autonomous vision detection in the field incorporated with GPS-INS navigation.

The effect of increasing the training sample through data augmentation for a large dataset to increase recall and precision has been identified as one of the future directions

for this study, as there are various data configuration and severity levels associated with EPTN components that can be incorporated to provide a more accurate benchmark for EPTN faults object detection libraries. And because this involves a massive, large drone image dataset, our future work also includes developing a process to automatically label millions of image tiles in a systematic way other than the manual labelling process that we went through in this study. The site inspection will also substantially benefit from automatic professional labelling of publicly available drone imagery for use in future deep learning object detection projects. Perhaps in the future, an image library of relevant faults will be available for improved computer vision techniques incorporating EPTN component maintenance. Natural habitat and generalizing flaws in varied ecosystems remain a challenge in terms of taxonomy, cause and effect, and severity levels, SSD FPN deep learning models in combination with more complicated data could potentially offer solutions.

### Abbreviations

CNNs	Convolutional neural networks
EPTN	Electric power transmission network
SSD	Single Shot Multibox Detector
FPN	Feature pyramid network
UAV	Unmanned area vehicle
VHR	Very High Resolution
SAR	Synthetic aperture radar
GSD	Ground sample data
DEM	Digital elevation model
RNN	Recurrent Neural Networks
ILN	Insulator Localizer Network
VGG	Visual Geometry Group
DDN	Defect Detector Network
RPN	Regional Proposal Network
RFCN	Region-based Fully Convolutional Network
RCNN	Region-based CNNs
RGB	Red, Green, Blue
IoU	Intersection over Union
mAP	Mean Average Precision
ReLU	Rectified Linear Units
SGD	Stochastic gradient descent
GAN	Generative Adversarial Networks
BN	Batch normalization
NMS	Non-maximum suppression

### Acknowledgements

We want to acknowledge Nigerian Electricity Commission (NEC) for giving us access to the dataset.

### Author contributions

IM and FIO developed the conceptual framework. CFI, JEA, OEO, GAC, and FE carried out the development and implementation under the supervision of IM and FIO. All the authors contributed to the writing of the paper. All authors read and approved the final manuscript.

### Funding

Not applicable.

### Availability of data and materials

The data is not available to the public because NEC allowed us to use the data on the condition that the data shall not be distributed or shared. The source code has been provided through this link: <https://github.com/Emekaking/Felect>.

### Declarations

#### Ethics approval and consent to participate

Not applicable.

#### Consent for publication

Not applicable.

**Competing interests**

There are no competing interests.

Received: 30 October 2021 Accepted: 27 May 2022

Published online: 17 June 2022

**References**

1. Nguyen VN, Jenssen R, Roverso D. Automatic autonomous vision-based power line inspection: a review of current status and the potential role of deep learning. *Int J Electr Power Energy Syst*. 2018;99:107–20. <https://doi.org/10.1016/j.jepes.2017.12.016>.
2. Kufeoglu S. Economic impacts of electric power outages and evaluation of customer interruption costs. Doctoral dissertation in permission of Aalto University, School of Electrical Engineering. 2015. p. 1–64.
3. Blimpo MP, Cosgrove-Davies M. Electricity access in sub-Saharan Africa: uptake, reliability, and complementary factors for economic impact. Africa Development Forum Washington, D.C. World Bank Group; 2019.
4. Ayodele T, Ogunjuyigbe A, Oladele O. Improving the transient stability of Nigerian 330Kv transmission network using static var compensation part I: the base study. *Niger J Technol*. 2015;35(1):155. <https://doi.org/10.4314/njt.v35i1.23>.
5. Bertheau P, Cader C, Blechinger P. Electrification modelling for Nigeria. *Energy Procedia*. 2016;93(March):108–12. <https://doi.org/10.1016/j.egypro.2016.07.157>.
6. Matikainen L, et al. Remote sensing methods for power line corridor surveys. *ISPRS J Photogramm Remote Sens*. 2016;119:10–31. <https://doi.org/10.1016/j.isprsjprs.2016.04.011>.
7. Xue Z, Luo S, Chen Y, Tong L. The application of the landslides detection method based on SAR images to transmission line corridor area. In: 2016 13th international computer conference on wavelet active media technology and information processing (ICCWAMTIP). 2017. p. 163–6. <https://doi.org/10.1109/ICCWAMTIP.2016.8079829>.
8. Yan L, Wu W, Li T. Power transmission tower monitoring technology based on TerraSAR-X products. In: International symposium on lidar and radar mapping 2011: technologies and applications, vol. 8286. 2011. p. 82861E. <https://doi.org/10.1117/12.912336>.
9. Luque-Vega LF, Castillo-Toledo B, Loukianov A, Gonzalez-Jimenez LE. Power line inspection via an unmanned aerial system based on the quadrotor helicopter. In: MELECON 2014–2014 17th IEEE Mediterranean electrotechnical conference. 2014. p. 393–7. <https://doi.org/10.1109/MELCON.2014.6820566>.
10. Wang M, Tong W, Liu S. Fault detection for power line based on convolution neural network. In: Proceedings of the 2017 international conference on deep learning technologies, vol. Part F1285. 2017. p. 95–101. <https://doi.org/10.1145/3094243.3094254>.
11. Guan H, Yu Y, Li J, Ji Z, Zhang Q. Extraction of power-transmission lines from vehicle-borne lidar data. *Int J Remote Sens*. 2016;37(1):229–47. <https://doi.org/10.1080/01431161.2015.1125549>.
12. Ahmad J, Malik AS, Abdullah MF, Kamel N, Xia L. A novel method for vegetation encroachment monitoring of transmission lines using a single 2D camera. *Pattern Anal Appl*. 2015;18(2):419–40. <https://doi.org/10.1007/s10044-014-0391-9>.
13. Yu X, et al. Comparison of laser and stereo optical, SAR and InSAR point clouds from air- and space-borne sources in the retrieval of forest inventory attributes. *Remote Sens*. 2015;7(12):15933–54. <https://doi.org/10.3390/rs71215809>.
14. Jaya Bharata Reddy M, Karthik Chandra B, Mohanta DK. Condition monitoring of 11 kV distribution system insulators incorporating complex imagery using combined DOST-SVM approach. *IEEE Trans Dielectr Electr Insul*. 2013;20(2):664–74. <https://doi.org/10.1109/TDEI.2013.6508770>.
15. Jiang J, Zhao L, Wang J, Liu Y, Tang M, Ji Z. The electrified insulator parameter measurement for flashover based on photogrammetric method. In: MIPPR 2011: multispectral image acquisition, processing, and analysis, vol. 8002. Bellingham: International Society for Optics and Photonics; 2011. p. 800211. <https://doi.org/10.1117/12.902054>.
16. Hu Y, Liu K. Inspection and monitoring technologies of transmission lines with remote sensing. New York: Academic Press; 2017. p. 257–64.
17. Zormpas A, Moirgiorgou K, Kalaitzakis K, Plokamakis GA, Partsinevelos P. Power transmission lines inspection using properly equipped unmanned aerial vehicle (UAV). In: 2018 IEEE international conference on imaging systems and techniques (IST). 2018. p. 1–5.
18. Liu X, Miao X, Jiang H, Chen J. Review of data analysis in vision inspection of power lines with an in-depth discussion of deep learning technology. 2020. p. 1–29. <https://doi.org/10.1016/j.arcontrol.2020.09.002>.
19. Yu X, Wu X, Luo C, Ren P. Deep learning in remote sensing scene classification: a data augmentation enhanced convolutional neural network framework. *GIScience Remote Sens*. 2017;54(5):741–58. <https://doi.org/10.1080/15481603.2017.1323377>.
20. Baştanlar Y, Özuysal M. Introduction to machine learning. *Methods Mol Biol*. 2014;1107:105–28. [https://doi.org/10.1007/978-1-62703-748-8\\_7](https://doi.org/10.1007/978-1-62703-748-8_7).
21. Siddiqui ZA, Park U. A drone based transmission line components inspection system with deep learning technique. *Energies*. 2020;13(13):1–24. <https://doi.org/10.3390/en13133348>.
22. Han J, et al. Search like an eagle: a cascaded model for insulator missing faults detection in aerial images. *Energies*. 2020;13(3):1–20. <https://doi.org/10.3390/en13030713>.
23. Song Y, et al. A vision-based method for the broken spacer detection. In: 2015 IEEE international conference on cyber technology in automation, control, and intelligent systems (CYBER). 2015. p. 715–9. <https://doi.org/10.1109/CYBER.2015.7288029>.
24. Zhai Y, Cheng H, Chen R, Yang Q, Li X. Multi-saliency aggregation-based approach for insulator flashover fault detection using aerial images. *Energies*. 2018;11(2):1–12. <https://doi.org/10.3390/en11020340>.
25. Zhai Y, Wang D, Zhang M, Wang J, Guo F. Fault detection of insulator based on saliency and adaptive morphology. *Multimed Tools Appl*. 2017;76(9):12051–64. <https://doi.org/10.1007/s11042-016-3981-2>.



26. Han J, et al. A method of insulator faults detection in aerial images for high-voltage transmission lines inspection. *Appl Sci*. 2019;9(10):1–22. <https://doi.org/10.3390/app9102009>.
27. Liu Y, et al. 2016 4th international conference on applied robotics for the power industry, CARPI 2016. 2016. p. 1–5.
28. Fu J, Shao G, Wu L, Liu L, Ji Z. Defect detection of line facility using hierarchical model with learning algorithm. *High Volt Eng*. 2017;43(1):266–75. <https://doi.org/10.13336/j.1003-6520.hve.20161227035>.
29. Mao T, et al. Defect recognition method based on HOG and SVM for drone inspection images of power transmission line. In: 2019 international conference on high performance big data and intelligent systems (HPBD&IS), No. 61701404. 2019. p. 254–7. <https://doi.org/10.1109/HPBDIS.2019.8735466>.
30. Zhao Z, Xu G, Qi Y, Liu N, Zhang T. Multi-patch deep features for power line insulator status classification from aerial images. In: 2016 international joint conference on neural networks (IJCNN), vol. 2016–October. 2016. p. 3187–94. <https://doi.org/10.1109/IJCNN.2016.7727606>.
31. Liu X, Jiang H, Chen J, Chen J, Zhuang S, Miao X. Insulator detection in aerial images based on faster regions with convolutional neural network. In: 2018 IEEE 14th international conference on control and automation (ICCA), vol. 2018–June. 2018. p. 1082–6. <https://doi.org/10.1109/ICCA.2018.8444172>.
32. 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;7:61797–810. <https://doi.org/10.1109/ACCESS.2019.2915985>.
33. Tao X, Zhang D, Wang Z, Liu X, Zhang H, Xu D. Detection of power line insulator defects using aerial images analyzed with convolutional neural networks. *IEEE Trans Syst Man Cybern Syst*. 2020;50(4):1486–98. <https://doi.org/10.1109/TSMC.2018.2871750>.
34. Ma L, Xu C, Zuo G, Bo B, Tao F. Detection method of insulator based on faster R-CNN. In: 2017 IEEE 7th annual international conference on CYBER technology in automation, control, and intelligent systems (CYBER). 2018. p. 1410–4. <https://doi.org/10.1109/CYBER.2017.8446155>.
35. Bai R, Cao H, Yu Y, Wang F, Dang W, Chu Z. Insulator fault recognition based on spatial pyramid pooling networks with transfer learning (match 2018). In: 2018 3rd international conference on advanced robotics and mechatronics (ICARM). 2019. p. 824–8. <https://doi.org/10.1109/ICARM.2018.8610720>.
36. Yang Y, Wang L, Wang Y, Mei X. Insulator self-shattering detection: a deep convolutional neural network approach. *Multimed Tools Appl*. 2019;78(8):10097–112. <https://doi.org/10.1007/s11042-018-6610-4>.
37. Guo M, Fang J, Jun T, Tan S. Detection algorithm of untwisted or broken strands of transmission line based on FRAC. In: 2019 3rd international conference on electronic information technology and computer engineering (EITCE), No. 17030901015. 2019. p. 632–6. <https://doi.org/10.1109/EITCE47263.2019.9094879>.
38. Chen J, Xu X, Dang H. Fault detection of insulators using second-order fully convolutional network model. *Math Probl Eng*. 2019. <https://doi.org/10.1155/2019/6397905>.
39. Liao GP, Yang GJ, Tong WT, Gao W, Lv FL, Gao D. Study on power line insulator defect detection via improved faster region-based convolutional neural network. In: 2019 IEEE 7th international conference on computer science and network technology (ICCSNT). 2019. p. 262–6. <https://doi.org/10.1109/ICCSNT47585.2019.8962497>.
40. Li J, Yan D, Luan K, Li Z, Liang H. Deep learning-based bird's nest detection on transmission lines using UAV imagery. *Appl Sci*. 2020;10(18):6147. <https://doi.org/10.3390/app10186147>.
41. Olawale T, Ad SA. Report on student industrial work experience scheme (SIWES) at transmission company of Nigeria (TCN) for the student industrial work experience scheme (SIWES), No. August 2018. 2018. <https://doi.org/10.13140/RG.2.2.35067.05929>.
42. Hu Y, Liu K, Mengqi C. Free vibration analysis of transmission lines based on the dynamic stiffness method. *Soc Open Sci*. 2019;6(3): 181354.
43. Francois C. Deep learning with python, vol. 53. Berkeley: Apress; 2019.
44. D. of C. S. U. Stanford. CS231n: convolutional neural networks for visual recognition. Stanford Vision and Learning Lab. <https://cs231n.github.io/convolutional-networks/>. Accessed 21 Nov 2020.
45. Krizhevsky A, Sutskever I, Hinton GE. Handbook of approximation algorithms and metaheuristics. In: ImageNet classification with deep convolutional neural networks. 2012. p. 1–1432. <https://doi.org/10.1201/9781420010749>.
46. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: 3rd international conference on learning representations, ICLR 2015. 2015. p. 1–14.
47. Yun JW. Deep residual learning for image recognition. *Enzyme Microb Technol*. 2015;19(2):107–17. [arXiv:1512.03385v1](https://arxiv.org/abs/1512.03385v1).
48. Rosebrock A. ImageNet: VGGNet, ResNet, inception, and xception with Keras. <https://www.pyimagesearch.com/2017/03/20/imagenet-vggnet-resnet-inception-xception-keras/>. Accessed 25 Nov 2020.
49. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H. Mobilenets: efficient convolutional neural networks for mobile vision applications. 2017. [arXiv:1704.04861](https://arxiv.org/abs/1704.04861).
50. Ren S, He K, Girshick R, Sun J. Faster R-CNN 2015. *Biol Conserv*. 2015;158:196–204.
51. Dai J, Li Y, He K, Sun J. R-FCN: object detection via region-based fully convolutional networks. 2016. [arXiv:1605.06409](https://arxiv.org/abs/1605.06409).
52. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, vol. 2016–Dec. 2016. p. 779–88. <https://doi.org/10.1109/CVPR.2016.91>.
53. Liu W, et al. SSD Net. Lecture notes in computer science (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9905. 2015. p. 21–37.
54. Huang J, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In: Speed/accuracy trade-offs for modern convolutional object detectors, vol. 2017–Jan. 2017. p. 3296–305. <https://doi.org/10.1109/CVPR.2017.351>.
55. Shi W, Bao S, Tan D. FFESD: an accurate and efficient single-shot detector for target detection. *Appl Sci*. 2019. <https://doi.org/10.3390/app9204276>.
56. O'Shea K, Nash R. An introduction to convolutional neural networks. 2015. p. 1–11. [arxiv: 1511.08458](https://arxiv.org/abs/1511.08458).
57. Tsang S. Review: FPN—feature pyramid network (object detection). Medium. 2019. <https://towardsdatascience.com/review-fpn-feature-pyramid-network-object-detection-262fc7482610>. Accessed 27 Jan 2022.
58. Shanmugamani R, Rahman AGA, Moore SM, Koganti N. Deep learning for computer vision: expert techniques to train advanced neural networks using Tensorflow and Keras. 1st ed. Birmingham: Packt Publishing Ltd; 2018.

59. Saha S. A comprehensive guide to convolutional neural networks—the ELI5 way. Medium. 2018. <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>. Accessed 15 Nov 2020.
60. Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning, vol. 1. 2015. p. 448–56.
61. Perin G, Picek S. On the influence of optimizers in deep learning-based side-channel analysis. Cryptol. ePrint Arch. no. Report 2020/977. 2020. p. 1–22. <https://eprint.iacr.org/2020/977>.
62. Manaswi NK. Deep learning with applications using python. Bangalore: Apress; 2018. p. 91–6. <https://doi.org/10.1007/978-1-4842-3516-4>.

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)

---