

Detection and Monitoring of Power Line Corridor From Satellite Imagery Using RetinaNet and K-Mean Clustering

FATHI MAHDI ELSIDDIG HAROUN¹, SITI NORATIQA MOHAMAD DEROS,
AND NORASHIDAH MD DIN¹, (Senior Member, IEEE)

Institute of Energy Infrastructure, Universiti Tenaga Nasional (UNITEN), Kajang 43000, Malaysia

Corresponding author: Fathi Mahdi Elsiddig Haroun (fathi.mahdi@uniten.edu.my)

This work was supported in part by the Innovation and Research Management Center, Universiti Tenaga Nasional (UNITEN), under the BOLD 2020 Grant Scheme Code RJ010517844/051, and in part by The Energy Sphere Satellite Laboratory, Universiti Tenaga Nasional.

ABSTRACT Monitoring of electrical transmission towers (TTs) is required to maintain the integrity of power lines. One major challenge is monitoring vegetation encroachment that can cause power interruption. Most of the current monitoring techniques use unmanned aerial vehicles (UAV) and airborne photography as an observation medium. However, these methods are expensive and not practical for monitoring wide areas. In this paper, we introduced a new method for monitoring the power line corridor from satellite imagery. The proposed method consists of two stages. In the first stage, we used the existing state-of-the-art RetinaNet deep learning (DL) model to detect the locations of the TTs from satellite imagery. A routing algorithm has been developed to create a path between every adjacent detected TT. In addition to the routing algorithm, a corridor identification algorithm has been established for extracting the power line corridor area. In the second stage, the k-mean clustering algorithm has been used to highlight the VE regions within the power line corridor area after converting the target satellite image into hue, saturation, and value (HSV) color space. The proposed monitoring system was able to detect TTs from satellite imagery with a mean average precision (mAP) of 72.45% for an Intersection of Union (IoU) threshold of 0.5 and 85.21% for IoU threshold of 0.3. Also, the monitoring system was able to successfully discriminate high- and low-density vegetation regions within the power line corridor area.

INDEX TERMS Deep learning, K-mean, satellite images, transmission tower detection.

I. INTRODUCTION

Electricity is of major importance to modern life, where a power outage can pose risk to livelihood and economy. Electrical transmission lines carry electrical energy to substations where the transmission lines form the basic infrastructure for the electrical power transmission system. The process of transmitting electrical energy from generating stations needs to be uninterrupted. However, several natural challenges such as forest fires and vegetation encroachment may cause interruptions that can lead to power outages [1]. There are many remote sensing methods which are used for monitoring transmission lines such as periodic manual inspection, usage of light detection and ranging data, synthetic aperture radar (SAR) data, aerial photography data from unmanned aerial vehicles (UAV), and airborne photography. Most of

these methods are time-consuming, expensive in relation to the coverage area, and unsuitable for rough terrains [2], [3].

Using high-resolution satellite imagery is a promising solution for monitoring power transmission lines. Satellite imagery provides data for a wide coverage area with relatively low cost compared with other optical remote sensing (ORS) methods. Although satellite images can cover wide areas, they may lack in data accuracy [4], [5]. Most of the previous works in power transmission lines detection and monitoring depend on high-resolution UAV and aerial images that cannot rapidly cover wide areas compared with satellite images [6]–[12]. The previous studies on transmission line monitoring and detection from ORS images can be categorized into four main categories:

- Vegetation index (VI)-based methods, which depend on the natural behavior of plants, where the near Infrared (NIR) spectrum is reflected and the red color

The associate editor coordinating the review of this manuscript and approving it for publication was Shiqi Wang.

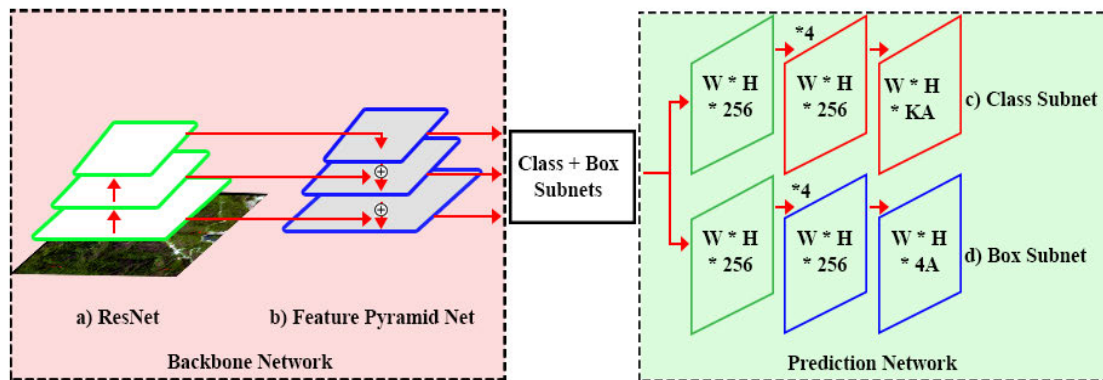


FIGURE 1. Illustration of RetinaNet structure that consists of two stages: the first stage is the backbone network, where (a) is the backbone CNN and (b) is the feature pyramid network. The second stage is the prediction network, where (c) is the classification subnet which consist of a FCN that detect the probability of observing a TT, and (d) is the regression subnet, which consist of a FCN that predict the anchors around the TT.

spectrum is absorbed. Although these methods can be effective for detecting vegetation encroachment, the VI-based methods have no ability to detect transmission towers (TTs). However, VI-based methods can be integrated with digital elevation modelling technology to detect the locations of the TTs [13]–[15].

- Stereo matching-based methods, where a 3D image is constructed from two different ORS images that are pointing to the same ground truth location. The main advantage of these methods is the ability to estimate the heights of objects surrounding the corridor area. Stereo matching-based methods depend on the availability of at least two satellite images from two different observation perspectives to form the disparity map [16]–[20].
- Object detection-based methods, where a set of image pre-processing steps need to be applied to detect the TTs locations and the corridor area. Object-detection based methods, involves manual setting of a set of filters for every different type of satellite image which make these methods ineffective [21], [22].
- Machine learning (ML)-based methods where most of these studies used ML algorithms for inspecting the integrity of the TT components and identifying possible failures from low altitude UAV images [23]–[31].

The automatic vegetation encroachment detection from satellite images have not been adequately studied. In this paper, we introduced a new automatic technique to monitor the power line corridor right-of-way from satellite imagery using the RetinaNet deep learning (DL) model and k-mean clustering.

II. BACKGROUND

This section provides an overview of the RetinaNet model and K-mean clustering algorithm.

A. RETINANET

RetinaNet, which is introduced by Lin *et al.* [32], has been mainly used to solve image detection and classification

problems. This DL model consists of two stages, as shown in Figure 1:

- The first stage is the backbone network, which consists of two components. The first component is the base convolutional neuron network (CNN), which reduces the size of the input image through several convolutional layers from bottom to top to generate the feature map. Many CNN can be used as a base model for RetinaNet, such as ResNet [33], VGG [34], and mobileNets [35]. The second component of the backbone network is the feature pyramid network (FPN) [36], where the last layer of the CNN will be up-sampled into higher scales from top to bottom to enable object detection at different sizes.
- The second stage is the prediction network, which consists of two components. The first component is the classification subnet which takes the combined output from every level of the FPN and predicts the output class k corresponding to every anchor A through a fully connected convolutional network (FCN) that has a width W and a height H . The second component is the regression subnet, which also consists of an FCN that takes the combined output of each level of the FPN and predict the anchor coordination $4A$ around the TT.

According to the original work of RetinaNet [32], the performance of the RetinaNet model was compared with other single-stage and two-stages state-of-the-art DL models such as Faster R-CNN++ [33], Faster R-CNN with FPN [36], Faster R-CNN by G-RMI [37], Faster R-CNN with TDM [38], YOLOv2 [39], SSD513 [40], [41], and DSSD513 [41], and the best mean average precision (mAP) achieved by RetinaNet with 59.1%, where the Intersection of Union (IoU) threshold was 0.5.

B. K-MEAN CLUSTERING

K-mean clustering is an unsupervised ML algorithm. The main idea behind k-mean clustering is to group different input data points based on the distance from the nearest centroid value, where the number of the desired centroids are related to the value of k . The automatic clustering process begins

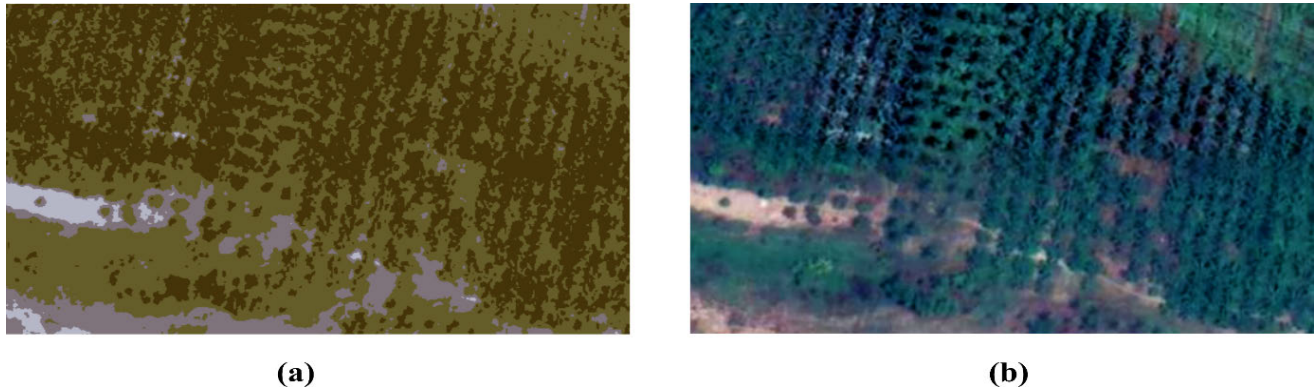


FIGURE 2. Image shows the difference between (a) k-mean clustered image where ($k = 5$) and (b) is the input image.

by setting an arbitrary centroid value; then, the location is adjusted so that each group of input data is associated with the nearest centroid [42]. The same approach also can be used to group similar neighbor values of image pixels into a unified pixel value, such as shown in Figure 2.

III. RELATED WORKS

ML algorithms have been implemented in a wide range of satellite imagery applications, such as land mapping and object detection [43]–[47]. However, the classification performance of the ML algorithms can be affected by the type of the extracted features, which can be a challenge in selecting the most related features. DL algorithms can be used to avoid this downside by taking the whole image as an input to the deep neuron network. This technique can automatically create a feature map through a set of deep CNN layers. The objects in satellite images are small in size with respect to the whole image, making objects detection localization in satellite images a challenging issue for ML algorithms. For instance, Malof *et al.* [48] used random forest ML classifier to detect small-scale solar photovoltaic panels from high-resolutions satellite images. The authors used a set of image pre-processing operations after the classification process to enhance the object detection and localization processes. The current state-of-the-art DL algorithms can solve both classification and localization problems that encouraged the choice of using DL algorithms for object detection in satellite imagery. Many works on using DL models for detecting small objects from satellite imagery showed promising results. For instance, Yun Ren *et al.* [49] have modified the R-CNN DL model to detect small objects in satellite imagery by reducing the size of anchors in the region proposal network (RPN) and adding a single high-level feature by designing a top-down and skip connection to the R-CNN architecture. The proposed modified R-CNN method achieved an average precision (AP) of 72.9% in ship detection.

Wang *et al.* [50] used RetinaNet to detect ships from SAR imagery. The study compared the performance of different DL models where the RetinaNet achieved the best mAP of 96.9% in comparison with SSD, Faster R-CNN and FPN.

One of the most used method to detect vegetation activity from satellite images is the normalized difference vegetation index where the vegetation observation process depends on the availability of the NIR band and the red green and blue (RGB) bands [51]–[53]. However, this method involves multispectral satellite images.

Several studies used the hue saturation and value (HSV) color space to directly detect the vegetation activity from the image pixels without using the multispectral satellite bands. For instance Hassanein *et al.* [54] used the hue color value as a main feature to discriminate between vegetation regions by utilizing only the RGB values. The proposed segmentation method achieved a mean accuracy of 87.29%. A similar study by Xiao *et al.* [55] proposed a vegetation segmentation algorithm based on the distribution of the hue channel and the roughness of the image which depends on the distribution of the saturation channel, where the proposed algorithm was able to identify vegetation areas from satellite images.

IV. METHODOLOGY

This section provides details about the proposed vegetation monitoring system. The monitoring system has been divided into two parts. The first part includes the TT detection, TT routing process, and corridor extraction process while the second part includes the K-mean clustering process.

A. DATASET CREATION AND LABELING

Training a DL model heavily depends on the amount of the prepared dataset. Ample data is required to ensure that the model will be able to handle the new cases.

To train, evaluate, and test the RetinaNet model, several satellite images containing transmission lines were collected from different sources, such as Google Map, ESRI Imagery, Google Earth, the Malaysian Space Agency (MYSA) and the electric transmission and distribution infrastructure imagery dataset [56]. The RGB color space bands of the dataset have been extracted and cropped into 1300×1300 pixel, where Figure 3 shows the histogram of the pixel intensity distribution of the dataset. The histogram distribution shows a very high values for the green pixels and lower values for red

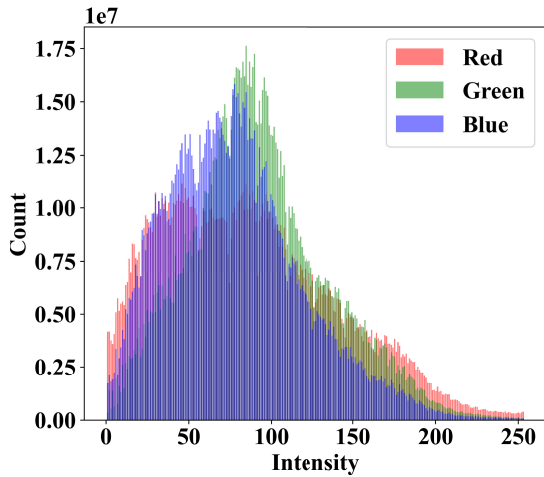


FIGURE 3. Histogram distribution of the dataset pixels intensity.

pixels which describes the behavior of the vegetated areas. The output of the labeling process was 2498 labels, where 2014 labels were extracted for training and 484 labels, were used for testing. Since the TTs are difficult to be observed from satellite image, the shadows of TTs are incorporated with their body structure and labeled as one unit to provide more information for the CNN.

B. TRAINING

After the labeling process, the RetinaNet model was trained with the ResNet-152 as a backbone for the CNN. The ResNet-152 was pretrained using the COCO dataset [57] to reduce the initial error loss value and the training time. The training process was performed using Fizyr Keras python framework [58], [59]. The total training period was about 18 hours and 41 minutes. The number of epochs was 200, and the minimum error loss was obtained at epoch number –148. A Graphical Processing Unit (GPU) was used to accelerate the training process. Table 1 shows the main specifications and configurations used in the training process, and Figure 4 shows classification loss, regression loss, and total loss per epoch. The derivation of the RetinaNet loss can be described by the following equations [32], [60]:

$$CE(p, y) = \begin{cases} -\log(p), & \text{if } y = 1 \\ -\log(1 - p), & \text{otherwise} \end{cases} \quad (1)$$

where CE is the cross-entropy loss, y is the class output and p represents the probability of the class where $p \in [0, 1]$.

The probability p of the output class y will be 1 for every positive sample. The same equation in (1) can be rewritten such as in (2):

$$p_t = \begin{cases} p, & \text{if } y = 1 \\ 1 - p, & \text{otherwise} \end{cases} \quad (2)$$

where $CE(p, y) = CE(p_t) = \log(p_t)$.

The final notation of the RetinaNet classification loss is the focal loss (FL) as described in equation (3):

$$FL(p_t) = -\alpha(1 - p_t)^\gamma \log(p_t) \quad (3)$$

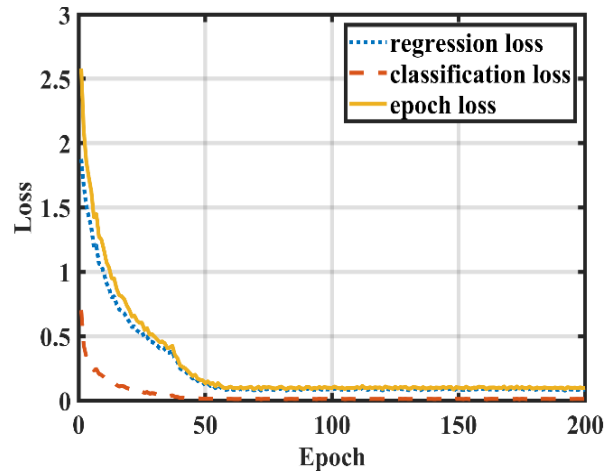


FIGURE 4. The regression loss, classification loss and total loss per epoch where the total loss is the algebraic summation of the regression and classification error loss.

TABLE 1. Main training specifications and configurations.

Description	Specification
Backbone model	ResNet-152
Training time	18 h, 40 m, 59 s
Number of epochs	200 epochs
Number of steps per epoch	400 steps
Learning rate	1e-5
Minimum total error loss	0.08679
Number of trainable parameters	70 8141 89
Input image size	1300 x 1300 pixels
Number of training samples	2014
Batch size	683 images
Data augmentation	True
Number of GPU units	1
GPU specification	NVIDIA GeForce GTX 1080 Ti[61]
CPU specification	AMD Ryzen Threadripper 1920X 12-Core Processor[62]
RAM size	64.0 GB

where α is a weight factor which is used to balance the output class, and γ is a focusing parameter where $\gamma \geq 0$.

On the other hand, the regression loss can be calculated as follows:

$$smooth_{L1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases} \quad (4)$$

$$L_{reg}(t^u, v) = \sum_{i \in \{x, y, w, h\}} smooth_{L1}(t_i^u - v_i) \quad (5)$$

where $smooth_{L1}(x)$ function was used to reduce the effect of the outliers, L_{reg} is the regression loss, t^u represent the prediction of the anchor, v is the required anchor value, and $\{x, y, w, h\}$ represents the anchor coordination. The final loss

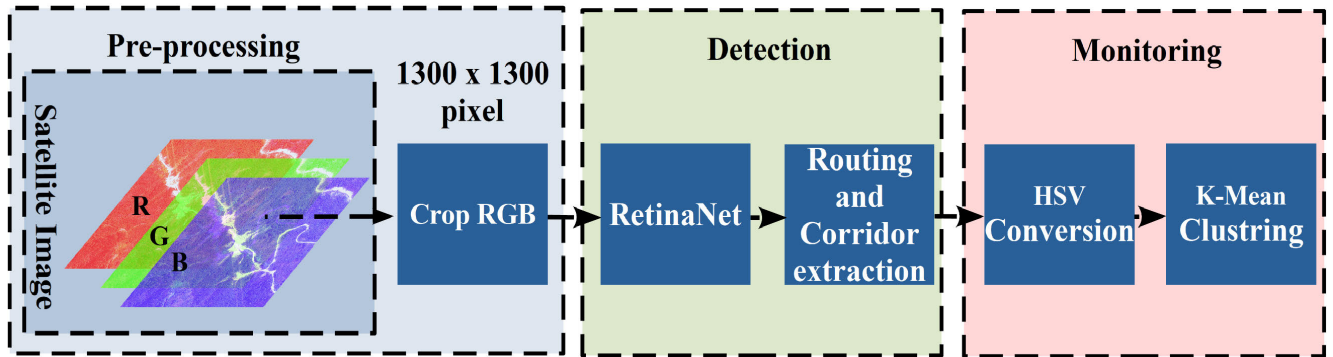


FIGURE 5. The proposed monitoring method that consists of three stages: satellite image preprocessing, transmission tower detection and vegetation encroachment monitoring.

L can be written as follows:

$$L = FL(p_t) + \lambda L_{reg}(t^u, v) \quad (6)$$

where λ is the control parameter of the loss balance. The overall power line monitoring method goes through several steps as shown in Figure 5. First of all, the size of the input image should be reduced to an appropriate size. Wherein this study, all the training and testing samples have been cropped into less than or equal to 1300×1300 pixels. In the second step, we used the trained RetinaNet model to predict the locations of the TTs. After the detection process a path creation algorithm has been used to create a path between every detected adjacent TTs. Before the monitoring step, a corridor extraction algorithm has been used to extract the power line corridor area from the background image.

C. TRANSMISSION TOWER ROUTING AND CORRIDOR EXTRACTION

The path of the power line can be predicted through the locations of the TTs especially, when the locations of the TTs form up a straight line. The purpose of identifying the power line path is to create a virtual path that describes the trajectory of the power line as illustrates in Figure 6. This process aimed to create a visual inspection mechanism that helps to observe whether the vegetation encroachment form a risk to the power lines.

Algorithm 1 provides pseudocode of the TT routing process, which automatically draws a line through all the TTs in the image in order to establish the path of the transmission line. The algorithm starts with the detection process in the while loop (line 1–4) to produce the bounding box (BB) coordination, where $box [b_0, b_1, b_2, b_3]$ is the coordination list. The centers of the BB are computed by finding the middle *point* (line 2) of the BB. Since the RetinaNet algorithm randomly detects TTs due to the FPN effect, the centers of all BBs are consequently sorted in ascending order according to their locations in the image (line 5) starting from location (0,0) to location (1300,1300). In the last part, the algorithm will draw lines between every adjacent center $draw_line((centres[i]), (centres[i + 1]))$ (line 10). Therefore, the final path between all TTs in the image will start from

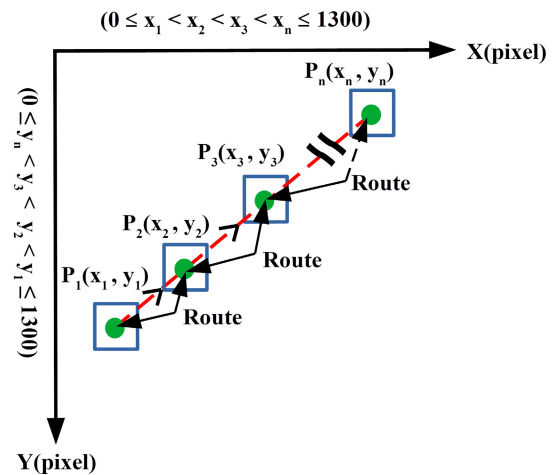


FIGURE 6. The process of creating the path between every adjacent centers in the image, where p represents the center of the BB.

Algorithm 1: TT routing

Input: The current detected BB that contains four elements stored in a list $box [b_0, b_1, b_2, b_3]$.
Output: Draw path that connects every BB centre.

1. **WHILE** detection is processing
2. $point \leftarrow [((box[b_0] + box[b_2])/2), ((box[b_1] + box[b_3])/2)]$
3. $centres \leftarrow centres[point]$
4. **END WHILE**
5. $sort(centres)$
6. **For** i in range($length(centres)$)
7. **IF** $(i+1) \geq length(centres)$
8. **BREAK**
9. **END IF**
10. $draw\ line((centres[i]), (centres[i+1]))$
11. **END FOR**
12. **END**

LISTING 1. TT routing algorithm which draws a line between every adjacent TT in the satellite image based on the spatial coordination order. The number of routing lines is $n - 1$ where n is the total number of detected TTs in the image.

the location with the least ranked to the highest ranked in the coordination list. However, if the TTs are closed to each other or there is a wrong classification of the TT, then the created path will be affected, and a wrong connection will be

Algorithm 2: Corridor detection

Input: The centre of every adjacent detected transmission towers As (x_1, y_1) and (x_2, y_2) and the guard distance m .

Output: Draw contour around the corridor area.

1. **WHILE** detection is processing
2. coordination of every adjacent TT $\leftarrow x_1, y_1, x_2, y_2$.
3. **END WHILE**
4. **For** j in range (length (number of TTs in the image))
5. **IF** $(j+1) \geq$ length (number of TTs in the image)
6. **BREAK**
7. **END IF**
8. Extract first centre point $(x_1, y_1) \leftarrow TT[j]$.
9. Extract second centre point $(x_2, y_2) \leftarrow TT[j+1]$.
10. Draw first parallel line \leftarrow Draw line $((x_1-m, y_1), (x_2-m, y_2))$.
11. Draw second parallel line \leftarrow Draw line $((x_1+m, y_1), (x_2+m, y_2))$.
12. **IF** $(j == 0)$
13. Draw first closing line \leftarrow Draw line $((x_1-m, y_1), (x_1+m, y_1))$.
14. **END IF**
15. **ELSE IF** $(j ==$ length (number of TTs in the image)-2)
16. Draw the second closing line \leftarrow Draw line $((x_2+m, y_2), (x_2-m, y_2))$.
17. **END IF**
18. Store left points \leftarrow left points. Append $((x_1-m, y_1))$.
19. Store left points \leftarrow left points. Append $((x_2-m, y_2))$.
20. Store right point \leftarrow right points. Append $((x_1+m, y_1))$.
21. Store right point \leftarrow right points. Append $((x_2+m, y_2))$.
22. **END FOR**
23. Store all right points in contour list \leftarrow contour. Append (right points).
24. Sort right points in a descending manner \leftarrow sort (right points).
25. Store all left points in contour list \leftarrow contour. Append (left points).
26. Draw the closed contour \leftarrow draw (contour).
27. **END**

LISTING 2. Corridor detection algorithm which detects the power line corridor area by establishing two lines parallel to the routing lines then drawing a closed surface that isolates the corridor area from the background image. The distance between the center of the TT and the parallel lines is m where m can be determined by the user based on the surrounding environment.

generated. The purpose of the corridor extraction process is to confine the monitoring process within the corridor area. The power line corridor detection process is vital to power line safety. For example, power line corridor detection can be used to inspect the integrity of the power lines [8].

Algorithm 2 is the pseudocode of the developed corridor extraction algorithm. The algorithm starts by getting all the centre points of the TTs (lines 1-3). Then, for every detected adjacent TTs pair in the image, the algorithm extracts the center coordination of the adjacent pairs (x_1, y_1) , (x_2, y_2) and draws two lines parallel to the routing path (lines 4-11) as

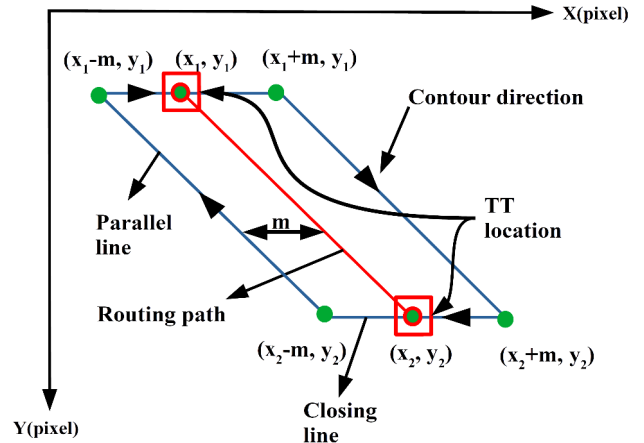


FIGURE 7. Corridor extraction process where (x_1, y_1) and (x_2, y_2) represent the centers of two detected adjacent TTs in the image (x_2-m, y_2) , (x_1-m, y_1) , (x_1+m, y_1) , and (x_2+m, y_2) are the contour coordination, where m is the guard distance.

shown in Figure 7. This aims to make the corridor detection process dynamic in which the corridor extraction algorithm can follow any sudden turns in the corridor path. The parallel lines represent the corridor border where m is the distance between the centre of the routing path and the surrounding environment. The distance m value governs by the type of the surrounding environment and can be predefined by the user.

In order to discover the closed corridor area, the algorithm draws two parallel lines (lines 12-17) to create a closed surface as shown in Figure 7. After determining the corridor area, the algorithm stores all the closed surface points (lines 18-21), sorts the points in a clockwise manner, and draws a contoured surface around the corridor area (lines 23-26).

This method is valid for all positions except in case if the centre points reside at the same alignment at 180° the image should be rotate at any positive or negative angle to create an offset along the x axis.

D. VEGETATION MONITORING

To detect vegetation along the power line corridor, a color clustering-based technique has been used to distinguish between high- and low-density vegetation regions. The monitoring process was directly applied after the detection process. The proposed monitoring technique depends on the properties of the HSV color space, where the HSV channels can differentiate the VE regions. The direct conversion from RGB to HSV color space, will produce a very sharp colors in some regions, as shown in Figure 8. Therefore, the k-mean clustering algorithm was used to reduce the sharp colors by staging the colors into only five groups.

Experimentally, using visual inspection, we found that the visual vegetation-dense discrimination will be more precise when the number of color clusters $k = 5$ and $k = 4$. The conversion from RGB to HSV can be described in the following equations [63]:

$$R' = \frac{R}{255}, \quad G' = \frac{G}{255}, \quad B' = \frac{B}{255} \quad (7)$$

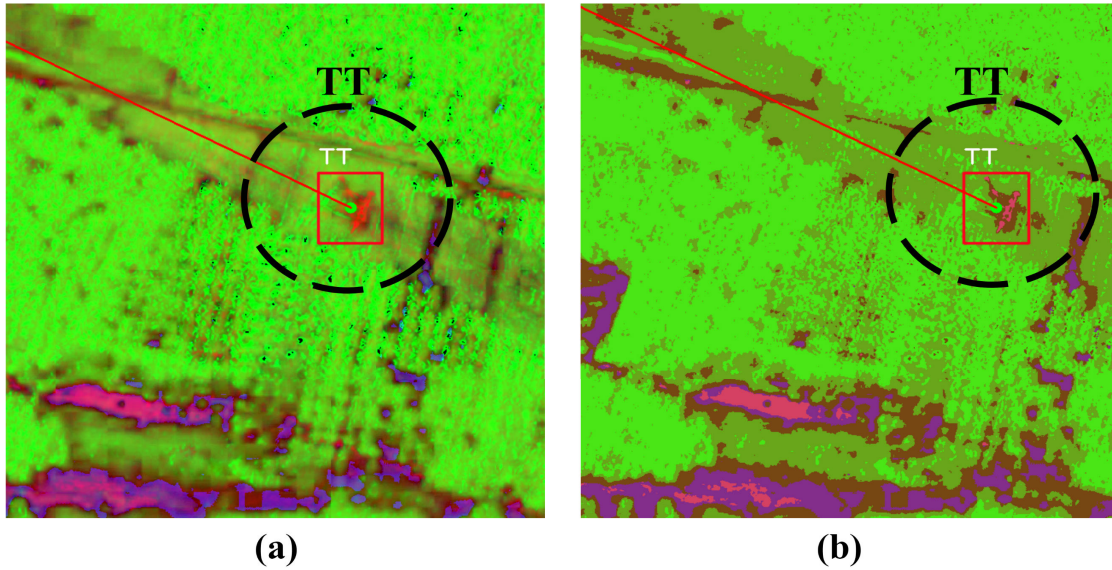


FIGURE 8. The difference between the direct conversion of the satellite image into HSV color space and the effect of using the k-mean clustering algorithm to reduce the sharp colors where (a) represents the direct HSV conversion of the satellite image and (b) represents the k-mean clustering of the HSV image where the number of color clusters ($k = 5$).

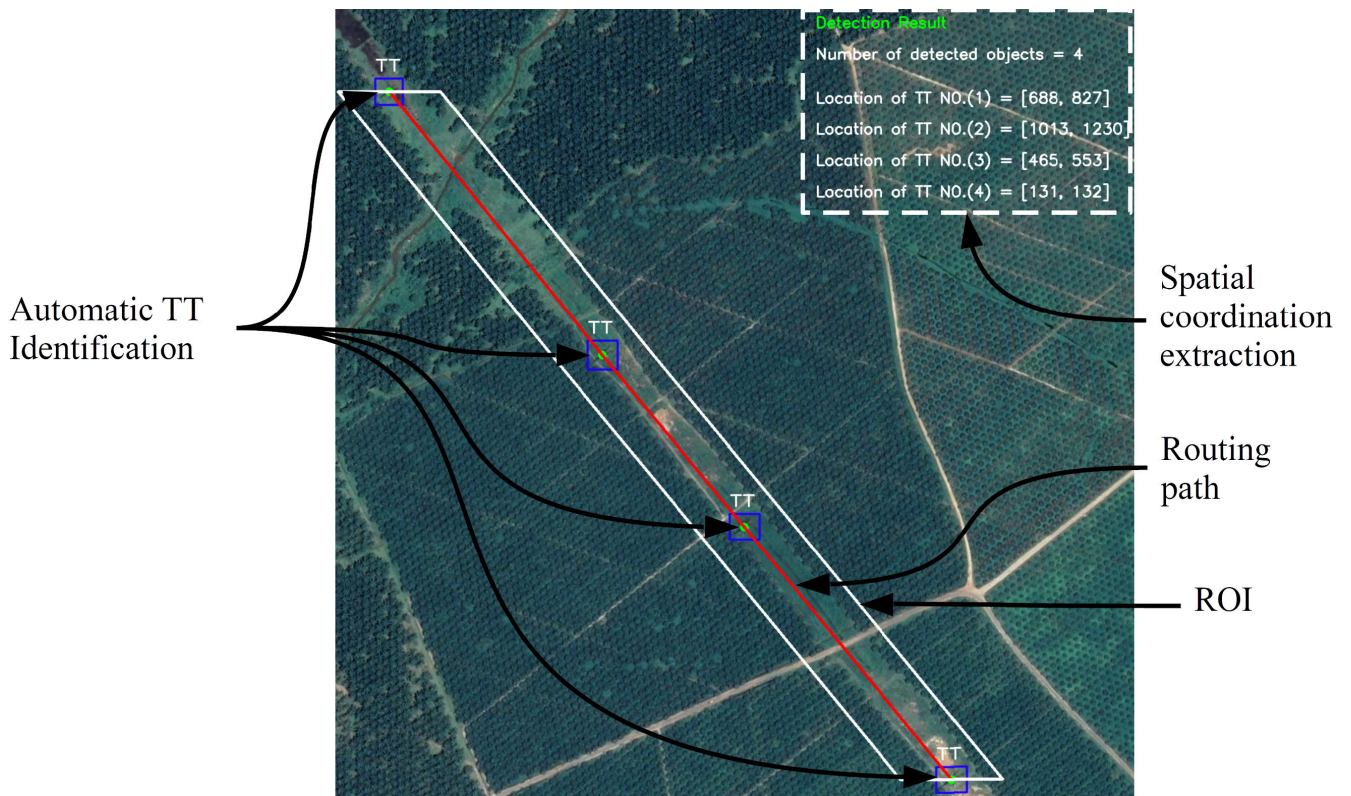


FIGURE 9. An example of the detection result where the proposed detection method was able to identify TTs from satellite image, extract the spatial coordination noting that the spatial coordination listing not in order, and to identify the transmission line right-of-way.

where R', G' and B' represent a rescaled RGB range that converts the color range from [0-255] to [0-1].

$$C_{max} = MAX (R', G', B'), C_{min} = MIN (R', G', B')$$

$$\Delta = C_{max} - C_{min} \tag{8}$$

$$H = \begin{cases} 60^\circ \times \left(\frac{G' - B'}{\Delta} \text{mod} (6) \right), & C_{max} = R' \\ 60^\circ \times \left(\frac{B' - R'}{\Delta} + 2 \right), & C_{max} = G' \\ 60^\circ \times \left(\frac{R' - G'}{\Delta} + 4 \right), & C_{max} = B' \end{cases} \tag{9}$$

$$S = \begin{cases} 0, & \Delta = 0 \\ \frac{\Delta}{C_{max}}, & \Delta \neq 0 \end{cases} \quad (10)$$

$$V = C_{max} \quad (11)$$

V. RESULTS

This section discusses about the evaluation results and the computation used to find the mAP. The evaluation test cases are 171 images that contain 484 TTs. The hardware specifications used in the training process were the same as those used in the evaluation process.

A. mAP

The mean distribution precision as known mAP is used to evaluate the performance of the detection model as in equation (16). The mAP can be calculated using the following equations (12) – (16) [64]–[66]:

$$p = \frac{\text{detected items}}{\text{detected items} + \text{undetected items}} \quad (12)$$

$$r = \frac{\text{detected items}}{\text{detected items} + \text{false detected items}} \quad (13)$$

The relation between precision p and the recall r can be formed in the shape of a curve where the recall accuracy divided into 11 equal intervals, and the area under the curve represents the AP as in equation (14):

$$AP = \frac{1}{11} \sum_{i \in \{0,0.1,0.2,\dots,1\}} P_{interp}(i) \quad (14)$$

The $P_{interp}(i)$ is the interpolation of the precision at the i instance of the recall where the interpolation can be calculated as in equation (15).

$$P_{interp}(i) = \max_{\tilde{r}, \tilde{r} \geq r} p(\tilde{i}) \quad (15)$$

where $p(\tilde{i})$ is the precision at a given \tilde{i} instance.

$$mAP = \frac{\sum_{i=1}^N AP(i)}{N} \quad (16)$$

where N is the total number of given samples and i is the instance sample.

B. INTERSECTION OF UNION

The IoU is the area shared between the ground truth BB and the predicted anchor over the total union area, as shown in equation (17), where A is the ground truth area and B is the predicted area [67].

$$IoU = \frac{A \cap B}{A \cup B} \quad (17)$$

The accuracy of the detection process depends on the threshold value of the BB.

TABLE 2. The mAP for different IoU threshold values.

IoU threshold	mAP	Inference time (s)
0.3	0.8521	0.1497
0.4	0.8205	0.1495
0.5	0.7245	0.1509
0.6	0.5362	0.1500

TABLE 3. Comparison between different state-of-art DL models.

Model	Backbone	mAP (IoU = 0.5)
YOLO V3	DarkNet	0.4570
SSD640	MobileNet	0.5872
Faster R-CNN	ResNet-50	0.6083
Faster R-CNN	ResNet-152	0.6491
RetinaNet	ResNet-152	0.7245

C. EVALUATION RESULTS

The evaluation of the TT detection model was performed on 171 images that contain 484 TTs, where a mAP of 0.7245 was obtained for $IoU \geq 0.5$ and 0.8521 for $IoU \geq 0.3$. Table 2 shows the outcome of the mAP and IoU and. The average inference time was 0.14718 seconds, where the inference time is the time taken by the hardware to drive the result of a new sample.

D. DETECTION RESULTS

After evaluating the performance of the TT detection model, the detection system was tested on several new samples where the system was able to achieve the following, as shown in Figure 9:

- Identifying the TTs in the image.
- Identifying the transmission line right-of-way.
- Extracting the spatial coordination for every TT in the image.

The performance of the RetinaNet has been compared with others state-of-the-art deep learning models such as YOLO, SSD, and Faster R-CNN as shown on Table 3. All the models were pretrained on the COCO standard dataset and fine-tuned using the custom TT dataset.

However, DL accelerators such as GPUs can generate false hardware error that can propagate to the software calculations and effect the accuracy of the model [68].

E. MONITORING RESULTS

The proposed monitoring system has been tested on several images that contain a power line corridor crossing area. The k-mean clustering of the HSV conversion showed a good ability to discriminate between high- and low-dense vegetation regions, as shown in Figure 10.

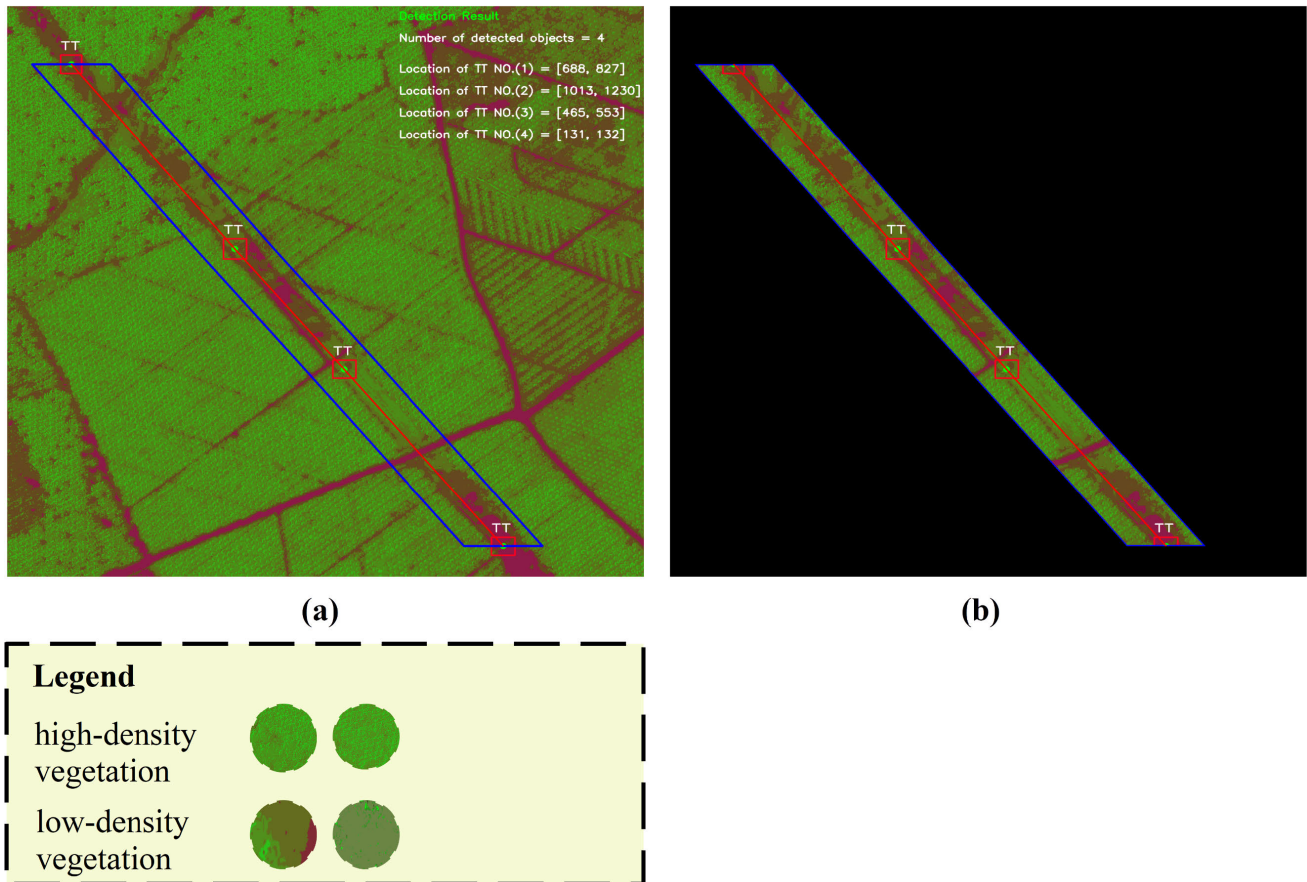


FIGURE 10. Examples of the vegetation monitoring results along the power line corridor, where (a) is the direct clustered HSV transformation of the target satellite image with ($k = 5$) and (b) is the extracted region of interest.

VI. CONCLUSION AND FUTURE WORK

In this paper, we introduced a new method of monitoring the power line corridor through satellite images using RetinaNet and k-mean clustering algorithms.

The proposed monitoring method can automatically detect TTs with a mAP of 0.7245 for IoU ≥ 0.5 .

Beside this, the monitoring system able track the power line where a routing algorithm has been developed to route all the TTs in the image. In addition to the routing algorithm, an automatic corridor extraction algorithm has been developed to extract the region of interest. We also have successfully shown that the monitoring of the VE near the TTs can be carried out by discriminating between high- with low low-density vegetation areas using the k-mean clustering algorithm on the HSV conversion of the satellite image.

This study has some limitations that can be improved in the future works. The suggested improvements are listed below.

- Improve the mAP by selecting the most related samples for training, which can also improve the false routing issue related to the false and miss classifications.
- Improve the routing algorithm to enable parallel line detection.
- Estimate the vegetation height, which can provide a better monitoring solution.

- Improve the monitoring method to enable monitoring in panchromatic satellite images.

ACKNOWLEDGMENT

The authors would like to thank The Energy Sphere Satellite Laboratory, Universiti Tenaga Nasional, for facilitating the equipment.

REFERENCES

- [1] J. Gomes-Mota, A. Brantes, L. C. Pinto, D. de Exploracao, and F. Azevedo, "How environmental factors impact line performance, field results from southwest Europe," in *Proc. 11th Int. Conf. Environ. Electr. Eng.*, May 2012, pp. 1053–1058, doi: [10.1109/EEEIC.2012.6221535](https://doi.org/10.1109/EEEIC.2012.6221535).
- [2] J. Ahmad, A. S. Malik, L. Xia, and N. Ashikin, "Vegetation encroachment monitoring for transmission lines right-of-ways: A survey," *Electr. Power Syst. Res.*, vol. 95, no. 1, pp. 339–352, Feb. 2013, doi: [10.1016/j.epsr.2012.07.015](https://doi.org/10.1016/j.epsr.2012.07.015).
- [3] Z. Li, R. Walker, R. Hayward, and L. Mejias, "Advances in vegetation management for power line corridor monitoring using aerial remote sensing techniques," in *Proc. 1st Int. Conf. Appl. Robot. Power Ind. (CARPI)*, Oct. 2010, pp. 1–6, doi: [10.1109/CARPI.2010.5624431](https://doi.org/10.1109/CARPI.2010.5624431).
- [4] S. Agrawal and G. B. Khairnar, "A comparative assessment of remote sensing imaging techniques: Optical, SAR and LiDAR," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. XLII-5/W3, pp. 1–6, Dec. 2019, doi: [10.5194/isprs-archives-XLII-5-W3-1-2019](https://doi.org/10.5194/isprs-archives-XLII-5-W3-1-2019).
- [5] J. Ahmad, A. Saeed Malik, and L. Xia, "Effective techniques for vegetation monitoring of transmission lines right-of-ways," in *Proc. IEEE Int. Conf. Imag. Syst. Techn.*, May 2011, pp. 34–38, doi: [10.1109/IST.2011.5962216](https://doi.org/10.1109/IST.2011.5962216).

- [6] Y. Zhang, X. Yuan, W. Li, and S. Chen, "Automatic power line inspection using UAV images," *Remote Sens.*, vol. 9, no. 8, p. 824, Aug. 2017, doi: [10.3390/rs9080824](https://doi.org/10.3390/rs9080824).
- [7] Y. Chen, Y. Li, H. Zhang, L. Tong, Y. Cao, and Z. Xue, "Automatic power line extraction from high resolution remote sensing imagery based on an improved radon transform," *Pattern Recognit.*, vol. 49, pp. 174–186, Jan. 2016, doi: [10.1016/j.patcog.2015.07.004](https://doi.org/10.1016/j.patcog.2015.07.004).
- [8] G. Józków, B. V. Jagt, and C. Toth, "Experiments with UAS imagery for automatic modeling of power line 3D geometry," *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. XL-1/W4, pp. 403–409, Aug. 2015, doi: [10.5194/isprsarchives-XL-1-W4-403-2015](https://doi.org/10.5194/isprsarchives-XL-1-W4-403-2015).
- [9] G. Yan, C. Li, G. Zhou, W. Zhang, and X. Li, "Automatic extraction of power lines from aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 3, pp. 387–391, Jul. 2007, doi: [10.1109/LGRS.2007.895714](https://doi.org/10.1109/LGRS.2007.895714).
- [10] M. Gerke and P. Seibold, "Visual inspection of power lines by U.A.S.," in *Proc. Int. Conf. Expo. Electr. Power Eng. (EPE)*, Oct. 2014, pp. 1077–1082, doi: [10.1109/ICEPE.2014.6970074](https://doi.org/10.1109/ICEPE.2014.6970074).
- [11] J. Zhang, L. Liu, B. Wang, X. Chen, Q. Wang, and T. Zheng, "High speed automatic power line detection and tracking for a UAV-based inspection," in *Proc. Int. Conf. Ind. Control Electron. Eng.*, Aug. 2012, pp. 266–269, doi: [10.1109/ICICEE.2012.77](https://doi.org/10.1109/ICICEE.2012.77).
- [12] H. Zhang, W. Yang, H. Yu, F. Xu, and H. Zhang, "Combined convolutional and structured features for power line detection in UAV images," in *Proc. Int. Geosci. Remote Sens. Symp.*, Jul. 2019, pp. 1306–1309, doi: [10.1109/IGARSS.2019.8898033](https://doi.org/10.1109/IGARSS.2019.8898033).
- [13] S. Mikhailap, S. Trashchenkov, and V. Vasilyeva, "Study of overhead power line corridors on the territory of pskov region (Russia) based on satellite sounding data," *Vide. Tehnol. Resur. Environ. Technol. Resour.*, vol. 1, pp. 164–167, Jun. 2019, doi: [10.17770/etr2019vol1.4087](https://doi.org/10.17770/etr2019vol1.4087).
- [14] M. G. M. Jardini, J. A. Jardini, L. C. Magrini, M. Masuda, P. R. le Silva, J. A. Quintanilha, A. M. K. Beltrame, and R. M. Jacobsen, "Vegetation control of transmission lines right-of-way," in *Proc. 3rd Int. Conf. Recent Adv. Space Technol.*, Jun. 2007, pp. 660–665, doi: [10.1109/RAST.2007.4284075](https://doi.org/10.1109/RAST.2007.4284075).
- [15] Y. Kobayashi, G. G. Karady, G. T. Heydt, and R. G. Olsen, "The utilization of satellite images to identify trees endangering transmission lines," *IEEE Trans. Power Del.*, vol. 24, no. 3, pp. 1703–1709, Jul. 2009, doi: [10.1109/TPWRD.2009.2022664](https://doi.org/10.1109/TPWRD.2009.2022664).
- [16] A. Qayyum, A. S. Malik, M. Naufal, M. Saad, M. Iqbal, R. F. Ahmad, and T. A. R. B. T. Abdullah, "Dynamic programming based comparison including QuickBird and IKONOS satellite stereo images for monitoring vegetation near power poles," in *Proc. IEEE Int. Conf. Control Syst., Comput. Eng. (ICCSCE)*, Nov. 2014, pp. 576–581, doi: [10.1109/ICCSCE.2014.7072784](https://doi.org/10.1109/ICCSCE.2014.7072784).
- [17] A. Qayyum, A. S. Malik, M. N. M. Saad, M. Iqbal, R. F. Ahmad, T. A. R. B. T. Abdullah, and A. Q. Ramli, "Monitoring of vegetation near power lines based on dynamic programming using satellite stereo images," in *Proc. IEEE Int. Conf. Smart Instrum., Meas. Appl. (ICSIMA)*, Nov. 2014, pp. 1–6, doi: [10.1109/ICSIMA.2014.7047429](https://doi.org/10.1109/ICSIMA.2014.7047429).
- [18] M. S. Moeller, "Monitoring powerline corridors with stereo satellite imagery," in *Proc. MAPPS/ASPRS Conf.*, 2006, pp. 1–6.
- [19] Z. Li, Y. Liu, R. Walker, R. Hayward, and J. Zhang, "Towards automatic power line detection for a UAV surveillance system using pulse coupled neural filter and an improved Hough transform," *Mach. Vis. Appl.*, vol. 21, no. 5, pp. 677–686, Aug. 2010, doi: [10.1007/s00138-009-0206-y](https://doi.org/10.1007/s00138-009-0206-y).
- [20] S. Collings, E. Botha, J. Anstee, and N. Campbell, "Depth from satellite images: Depth retrieval using a stereo and radiative transfer-based hybrid method," *Remote Sens.*, vol. 10, no. 8, p. 1247, Aug. 2018, doi: [10.3390/rs10081247](https://doi.org/10.3390/rs10081247).
- [21] A. Zain, "Transmission lines monitoring from satellite images," *J. Mech. Continua Math. Sci.*, vol. 13, no. 5, pp. 112–121, Oct. 2018, doi: [10.26782/jmcsms.2018.10.00010](https://doi.org/10.26782/jmcsms.2018.10.00010).
- [22] J. Ahmad, A. S. Malik, M. F. Abdullah, N. Kamel, and L. Xia, "A novel method for vegetation encroachment monitoring of transmission lines using a single 2D camera," *Pattern Anal. Appl.*, vol. 18, no. 2, pp. 419–440, May 2015, doi: [10.1007/s10044-014-0391-9](https://doi.org/10.1007/s10044-014-0391-9).
- [23] V. N. Nguyen, R. Jenssen, and D. Roverso, "LS-Net: Fast single-shot line-segment detector," *Mach. Vis. Appl.*, vol. 32, no. 1, pp. 1–16, 2021, doi: [10.1007/s00138-020-01138-6](https://doi.org/10.1007/s00138-020-01138-6).
- [24] K. Sikorska-Lukasiewicz, "Methods of automatic vegetation encroachment detection for high voltage power lines," in *Proc. Radioelectron. Syst. Conf.*, Feb. 2020, p. 63, doi: [10.1117/12.2565756](https://doi.org/10.1117/12.2565756).
- [25] E. Titov, O. Limanovskaya, A. Lemekh, and D. Volkova, "The deep learning based power line defect detection system built on data collected by the cablewalker drone," in *Proc. Int. Multi-Conf. Eng., Comput. Inf. Sci. (SIBIRCON)*, Oct. 2019, pp. 700–704, doi: [10.1109/SIBIRCON48586.2019.8958397](https://doi.org/10.1109/SIBIRCON48586.2019.8958397).
- [26] J. Gubbi, A. Varghese, and P. Balamuralidhar, "A new deep learning architecture for detection of long linear infrastructure," in *Proc. 15th IAPR Int. Conf. Mach. Vis. Appl. (MVA)*, May 2017, pp. 207–210, doi: [10.23919/MVA.2017.7986837](https://doi.org/10.23919/MVA.2017.7986837).
- [27] C. Sampedro, C. Martinez, A. Chauhan, and P. Campoy, "A supervised approach to electric tower detection and classification for power line inspection," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2014, pp. 1970–1977, doi: [10.1109/IJCNN.2014.6889836](https://doi.org/10.1109/IJCNN.2014.6889836).
- [28] V. N. Nguyen, R. Jenssen, and D. Roverso, "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.*, vol. 99, pp. 107–120, Jul. 2018, doi: [10.1016/j.ijepes.2017.12.016](https://doi.org/10.1016/j.ijepes.2017.12.016).
- [29] V. N. Nguyen, R. Jenssen, and D. Roverso, "Intelligent monitoring and inspection of power line components powered by UAVs and deep learning," *IEEE Power Energy Technol. Soc. J.*, vol. 6, no. 1, pp. 11–21, Mar. 2019, doi: [10.1109/jpets.2018.2881429](https://doi.org/10.1109/jpets.2018.2881429).
- [30] Z. A. Siddiqui, U. Park, S. W. Lee, N. J. Jung, M. Choi, C. Lim, and J. H. Seo, "Robust powerline equipment inspection system based on a convolutional neural network," *Sensors*, vol. 18, no. 11, pp. 1–25, 2018, doi: [10.3390/s18113837](https://doi.org/10.3390/s18113837).
- [31] Z. Hu, T. He, Y. Zeng, X. Luo, J. Wang, S. Huang, J. Liang, Q. Sun, H. Xu, and B. Lin, "Fast image recognition of transmission tower based on big data," *Protection Control Modern Power Syst.*, vol. 3, no. 1, pp. 1–10, Dec. 2018, doi: [10.1186/s41601-018-0088-y](https://doi.org/10.1186/s41601-018-0088-y).
- [32] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007, doi: [10.1109/ICCV.2017.324](https://doi.org/10.1109/ICCV.2017.324).
- [33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778, doi: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90).
- [34] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR) Track*, 2015, pp. 1–14.
- [35] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: <http://arxiv.org/abs/1704.04861>
- [36] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944, doi: [10.1109/CVPR.2017.106](https://doi.org/10.1109/CVPR.2017.106).
- [37] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, "Speed/Accuracy trade-offs for modern convolutional object detectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3296–3305, doi: [10.1109/CVPR.2017.351](https://doi.org/10.1109/CVPR.2017.351).
- [38] A. Shrivastava, R. Sukthankar, J. Malik, and A. Gupta, "Beyond skip connections: Top-down modulation for object detection," 2016, *arXiv:1612.06851*. [Online]. Available: <https://arxiv.org/abs/1612.06851>
- [39] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. 30th IEEE Conf. Comput. Vis. Pattern Recognit., (CVPR)*, Jan. 2017, pp. 6517–6525, doi: [10.1109/CVPR.2017.690](https://doi.org/10.1109/CVPR.2017.690).
- [40] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV) in Lecture Notes in Computer Science*, vol. 9905. Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 21–37, doi: [10.1007/978-3-319-46448-0_2](https://doi.org/10.1007/978-3-319-46448-0_2).
- [41] C. Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," 2017, *arXiv:1701.06659*. [Online]. Available: <https://arxiv.org/abs/1701.06659>
- [42] V. K. Dehariya, S. K. Shrivastava, and R. C. Jain, "Clustering of image data set using K-means and fuzzy K-means algorithms," in *Proc. Int. Conf. Comput. Intell. Commun. Netw.*, Nov. 2010, pp. 386–391, doi: [10.1109/CICN.2010.80](https://doi.org/10.1109/CICN.2010.80).
- [43] X. Luo, X. Tong, R. Wang, and H. Pan, "Evaluation of machine learning-based urban surface mapping using a new moderate-resolution satellite imagery dataset," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2019, pp. 9819–9822, doi: [10.1109/IGARSS.2019.8898065](https://doi.org/10.1109/IGARSS.2019.8898065).

- [44] J. Pasquet, T. Desert, O. Bartoli, M. Chaumont, C. Delenne, G. Subsol, M. Derras, and N. Chahinian, "Detection of manhole covers in high-resolution aerial images of urban areas by combining two methods," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 5, pp. 1802–1807, May 2016, doi: [10.1109/JSTARS.2015.2504401](https://doi.org/10.1109/JSTARS.2015.2504401).
- [45] H. Ma, N. Lu, L. Ge, Q. Li, X. You, and X. Li, "Automatic road damage detection using high-resolution satellite images and road maps," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2013, pp. 3718–3721, doi: [10.1109/IGARSS.2013.6723638](https://doi.org/10.1109/IGARSS.2013.6723638).
- [46] R. Avudaiamma, S. Dayana, R. Prabhu, and A. Swarnalatha, "Automatic building extraction from VHR satellite image," in *Proc. Int. Conf. Current Trends towards Converging Technol. (ICCTCT)*, Mar. 2018, pp. 1–6, doi: [10.1109/ICCTCT.2018.8551119](https://doi.org/10.1109/ICCTCT.2018.8551119).
- [47] M. A. Zurbaran, P. Wightman, and M. A. Brovelli, "A machine learning pipeline articulating satellite imagery and openstreetmap for road detection," *ISPRS Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. XLII-4/W14, pp. 255–260, Aug. 2019, doi: [10.5194/isprs-archives-XLII-4-W14-255-2019](https://doi.org/10.5194/isprs-archives-XLII-4-W14-255-2019).
- [48] J. M. Malof, K. Bradbury, L. M. Collins, and R. G. Newell, "Automatic detection of solar photovoltaic arrays in high resolution aerial imagery," *Appl. Energy*, vol. 183, pp. 229–240, Dec. 2016, doi: [10.1016/j.apenergy.2016.08.191](https://doi.org/10.1016/j.apenergy.2016.08.191).
- [49] Y. Ren, C. Zhu, and S. Xiao, "Small object detection in optical remote sensing images via modified faster R-CNN," *Appl. Sci.*, vol. 8, no. 5, p. 813, May 2018, doi: [10.3390/app8050813](https://doi.org/10.3390/app8050813).
- [50] Y. Wang, C. Wang, H. Zhang, Y. Dong, and S. Wei, "Automatic ship detection based on RetinaNet using multi-resolution Gaofen-3 imagery," *Remote Sens.*, vol. 11, no. 5, pp. 1–14, 2019, doi: [10.3390/rs11050531](https://doi.org/10.3390/rs11050531).
- [51] B. L. Bhavana, N. Sridevi, and R. Hebbar, "Tree crown detection and extraction from high resolution satellite images in an urban area," in *Proc. Int. Conf. Design Innov. 3Cs Compute Communicate Control (ICD3C)*, Apr. 2018, pp. 8–11, doi: [10.1109/ICD3C.2018.00011](https://doi.org/10.1109/ICD3C.2018.00011).
- [52] G. Singhal, B. Bansod, L. Mathew, and S. Taneja, "A preparatory comparison of landsat and sentinel satellite data for estimation of chlorophyll concentration in vegetation," in *Proc. 2nd Int. Conf. Micro-Electron. Telecommun. Eng. (ICMETE)*, Sep. 2018, pp. 230–233, doi: [10.1109/ICMETE.2018.00058](https://doi.org/10.1109/ICMETE.2018.00058).
- [53] J. Khan, P. Wang, Y. Xie, L. Wang, and L. Li, "Mapping MODIS LST NDVI imagery for drought monitoring in Punjab Pakistan," *IEEE Access*, vol. 6, pp. 19898–19911, 2018, doi: [10.1109/ACCESS.2018.2821717](https://doi.org/10.1109/ACCESS.2018.2821717).
- [54] M. Hassanein, Z. Lari, and N. El-Sheimy, "A new vegetation segmentation approach for cropped fields based on threshold detection from hue histograms," *Sensors*, vol. 18, no. 4, pp. 1–25, 2018, doi: [10.3390/s18041253](https://doi.org/10.3390/s18041253).
- [55] H. Xiao, C. Tong, and Q. Liu, "A new method for discovery of vegetation changes based on satellite ground photographs," in *Proc. 8th Int. Congr. Image Signal Process. (CISP)*, Oct. 2015, pp. 851–855, doi: [10.1109/CISP.2015.7407996](https://doi.org/10.1109/CISP.2015.7407996).
- [56] X. Bradbury, K. Han, Q. Nair, V. Pathirathna, and T. You. (2018). *Electric Transmission and Distribution Infrastructure Imagery Dataset*. Accessed: Aug. 14, 2020. [Online]. Available: <https://figshare.com/articles/>, doi: [10.6084/m9.figshare.6931088.v1](https://doi.org/10.6084/m9.figshare.6931088.v1).
- [57] T. Y. Lin et al., "Microsoft COCO: Common objects in context," in *Computer Vision—ECCV 2014 (Lecture Notes in Computer Science)*, vol. 8693, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, doi: [10.1007/978-3-319-10602-1_48](https://doi.org/10.1007/978-3-319-10602-1_48).
- [58] *Keras-Retinanet*. Accessed: Aug. 15, 2020. [Online]. Available: <https://github.com/fizyr/keras-retinanet>
- [59] *GitHub Fizyr/Keras-Retinanet: Keras Implementation of RetinaNet Object Detection*. Accessed: Apr. 30, 2021. [Online]. Available: <https://github.com/fizyr/keras-retinanet>
- [60] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448, doi: [10.1109/ICCV.2015.169](https://doi.org/10.1109/ICCV.2015.169).
- [61] *GeForce GTX 1080 Ti Graphics Cards | NVIDIA GeForce*. Accessed: Aug. 16, 2020. [Online]. Available: <https://www.nvidia.com/en-us/geforce/products/10series/geforce-gtx-1080-ti/>
- [62] *Ryzen™ Threadripper™ 1920X Processor | AMD*. Accessed: Aug. 16, 2020. [Online]. Available: <https://www.amd.com/en/products/cpu/amd-ryzen-threadripper-1920x>
- [63] G. Saravanan, G. Yamuna, and S. Nandhini, "Real time implementation of RGB to HSV/HSI/HSL and its reverse color space models," in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, Apr. 2016, pp. 462–466, doi: [10.1109/ICCSP.2016.7754179](https://doi.org/10.1109/ICCSP.2016.7754179).
- [64] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009, doi: [10.1007/s11263-009-0275-4](https://doi.org/10.1007/s11263-009-0275-4).
- [65] A. Tharwat, "Classification assessment methods," *Appl. Comput. Informat.*, vol. 17, no. 1, pp. 168–192, Jan. 2021, doi: [10.1016/j.aci.2018.08.003](https://doi.org/10.1016/j.aci.2018.08.003).
- [66] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2011.
- [67] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 658–666, doi: [10.1109/CVPR.2019.00075](https://doi.org/10.1109/CVPR.2019.00075).
- [68] K. Adam, I. I. Mohamed, and Y. Ibrahim, "A selective mitigation technique of soft errors for DNN models used in healthcare applications: DenseNet201 case study," *IEEE Access*, vol. 9, pp. 65803–65823, 2021, doi: [10.1109/ACCESS.2021.3076716](https://doi.org/10.1109/ACCESS.2021.3076716).



FATHI MAHDI ELSIDDIG HAROUN was born in Sana'a, Yemen, in 1996. He received the B.Sc. degree (Hons.) in electronics engineering from Sudan University of Science and Technology (SUST), Khartoum, Sudan, in 2018. He is currently pursuing the master's degree with Universiti Tenaga Nasional (UNITEN), Kajang, Malaysia. His research interest includes deep learning implementation on satellite images.



SITI NORATIQA MOHAMAD DEROS received the Bachelor of Marine Technology degree and the Ph.D. degree in science from Universiti Teknologi MARA, Malaysia. She is currently a Postdoctoral Researcher at the Institute of Energy Infrastructure (IEI), Universiti Tenaga Nasional (UNITEN). She is also currently working on object automated detection from satellite image processing. She has published several articles in the respective field in prestigious journals. Her main research interests include satellite remote sensing technology and geographic information systems (GISs) applied in environment and natural resources discovery.



NORASHIDAH MD DIN (Senior Member, IEEE) received the B.S. degree in electrical engineering from Memphis State University, USA, in 1985, and the M.S. and Ph.D. degrees in electrical engineering from Universiti Teknologi Malaysia, in 1989 and 2007, respectively. She is currently a Professor of electronics engineering at Universiti Tenaga Nasional, Malaysia. She has 30 years of academic and industry experience. Her research interests include communications networks, wide-area monitoring of electrical transmission towers, and low power communications for the Internet of Things. She is a registered Professional Engineer in Malaysia.