2022 FORTEI-International Conference on Electrical Engineering (FORTEI-ICEE)

Deep Learning Implementation for Snail Trails Detection in Photovoltaic Module

Fitriyanty Dwi Lestary
Department of Electrical Engineering
Universitas Hasanuddin
Makassar, Indonesia
fitriyantydwi@gmail.com

Syafaruddin
Department of Electrical Engineering
Universitas Hasanuddin
Makassar, Indonesia
syafaruddin@unhas.ac.id

Intan Sari Areni
Department of Electrical Engineering
Universitas Hasanuddin
Makassar, Indonesia
intan@unhas.ac.id

2022 FORTEI-International Conference on Electrical Engineering (FORTEI-ICEE) | 979-8-3503-9798-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/FORTEI-ICEE57243.2022.9972952

***Abstract*— The degradation of fossil fuel reserves and the increasing enthusiasm for the development of renewable resources have led the world to seek and create renewable resources. Among the alternative energy sources available, solar energy is claimed to be the main choice that can replace fossil resources to face the demands of power consumption needs at this time due to the unlimited supply of solar energy. However, the awareness to check the condition of Photovoltaic modules is still low. Detecting the snail trails is necessary to determine the performance of snail trails as an initial step to prevent ongoing damage. This study aims to develop a snail trails detection system on photovoltaic module images using the machine learning method called as deep learning with the YOLO (You Only Look Once) algorithm version 3. There are several important processes required in the YOLOv3 method, which are image annotation, training data, and testing data. The result of detecting snail trail objects on the photovoltaic module obtained 99.7% of accuracy. This value suggests that this method can be proposed to contribute to the implementation of the development of effective damage detection research early on solar panel modules.**

Keywords— Photovoltaic module, snail trails detection, image processing, YOLOv3

I. INTRODUCTION

The degradation of fossil fuel reserves and the increasing enthusiasm for the development of renewable resources have led the world to seek and create renewable resources. Among the alternative energy sources available, solar energy is claimed to be the main choice that can replace fossil resources to face the demands of power consumption needs at this time due to the unlimited supply of solar energy. But over time, the cost of producing PV cells is always decreased, causing the electricity production costs will approach the cost of conventional fuels[1].

Decrease in performance is one of the obstacles in the use of solar cells, is the quality and availability of power. This is influenced by low efficiency, due to events that cannot be avoided under external conditions. The causes of these external factors greatly affect the continuous operation of the photovoltaic module's lifetime. So it is important to ensure monitoring of the detection of faults that occur in the photovoltaic system in a timely manner. The decrease in the performance of the solar cell module is caused by three main factors, namely damage to the solar panel as a power absorber. So it is important to ensure monitoring of the detection of faults that occur in the photovoltaic system in a timely manner. The decrease in the performance of solar cell modules is caused by three main factors, namely physical damage (which can occur during the production and installation process), damage due to electrical circuits, and damage due to the influence of environmental conditions such as thermal stress, humidity, and temperature. Among these factors, interference due to physical damage and environmental conditions is the most frequent damage following the lifetime of the solar cell module. One of the physical damages caused by the environment is causing the appearance of fractures and snail trails on the surface of the photovoltaic module [2]. Snail trails form as small dark lines that are narrow and color change on the cell surface [3]. This snail trail phenomenon appears as a whitish, slightly brownish contact finger on the surface of the solar cell, especially at the edge of the cell or around the micro-crack area. The influence of micro-cracks on the surface of solar panels is the initial cause of the appearance of snail trails [4].

In a study conducted by Han-Chang Li, et al they used IEC61215 acceleration testing, to be able to find out the snail trail development of simulations created on a commercial scale [4]. The test successfully showed the beginning of the appearance of micro-cracks marked by the appearance of snail trails through electroluminescence testing. The use of this method is widely used and is very effective for detecting damage to solar modules, but the cost of using these tests is still relatively high and has not been widely provided in public laboratories.

The impact of technological advances creates a variety of techniques that have similarities that are effective and non-destructive in detecting a decrease in the work performance of solar modules such as electrical power measurements, thermography measurements, measurements using electroluminescence imaging, and measurements using photoluminescence imaging. The Deep Learning method is considered one of the advanced methods that can be used to solve problems with feature extraction and image classification in areas such as robotics, automation industries, and healthcare. The application of deep learning methods has the same high accuracy as conventional methods. Many learning models about deep learning methods that have been developed can be used for the automatic detection of damage to photovoltaic cells through an image. *Convolutional Neural Network* (CNN) is a method based on the VGG16 architecture that can be used to detect damaged cells by adopting an artificial neural network from an image of a solar panel surface captured to detect defects on the surface of a photovoltaic module. This approach is carried out by extracting significant features from *electroluminescence* (EL) images from PV cells or by drawing the surface of solar panels and using random forest classification algorithms [5].

Researchers in recent years have proposed various artificial neural network learning methods, one of which is the You Only Look Once (YOLO) method [6]. The YOLO method uses the Convolutional Neural Network (CNN) algorithm in its architecture and has given significant results in image and video identification. With these technological advances, it makes it easier to monitor and perform solar panels, so this research is useful in preventing more damage to the use of solar panels due to snail trails.

II. Materials and Methods

At the beginning of the research, the YOLO series algorithm was a target recognition method that used regression to connect between a variable bound to one another, proposed by Redmon et al [6]. With many advantages of the YOLO method, in 2018, YOLO has been developed with better performance to become the third generation, namely YOLOv3, response speed in detecting objects is faster, especially in dense and many images. YOLOv3 uses a multi-scale prediction method to increase the advantage of the YOLOv2 defect which still has accuracy for small target recognition. With the development of the third generation, YOLOv3 has a higher detection accuracy and a fast speed in processing object detection. In this study, the detection and classification of objects on the surface of the solar panel was carried out, namely the condition of the surface of the solar panel with snail trails or solar panels in good condition using the YOLO (You Only Look Once) method. YOLO uses a single Convolutional Neural Network (CNN) for object classification and localization using bounding boxes. The Convolutional Neural Network (CNN) algorithm method is widely used to view objects in the image. The YOLO architecture is simple and very fast in detecting objects, making the YOLO method researchers solve the problem of identifying snail trail objects on the surface of solar cells. In Table 1 it can be seen that the mAP (mean Average Precision) on YOLO is higher than other detection algorithms, meaning that the accuracy that will be obtained if using YOLO will be better when compared to other algorithms [7].

TABLE I. Comparisson of Detection Peformance on Coco Dataset

Model	*mAP*	*Flops*	*FPS*
SSD500	46,5	-	19
YOLOv2 608×608	48,1	62,94 Bn	40
Tiny YOLO	23,7	5,41 Bn	244
SSD321	45,4	-	16
R-FCN	51,9	-	12
Retinanet-101-800	57,5	-	5
YOLOv3-608	57,9	140,69 Bn	20

From Table 1 we can see a comparison of the existing YOLO models. YOLOv3 is a model that has the highest accuracy, of the previous model. Most detection systems first use classification to perform detection by applying the model to images in several locations and assigning confidence values to the appropriate image as detection material. YOLO is very different, YOLO uses an algorithm by applying a single neural network to the entire image pixel. This network will function to divide the image into several areas and then predict the bounding box and probability, for each bounding area box by considering the probability to classify that object as a target. Fig. 1 shows the yolov3 network structure [8]. First, YOLOv3 will scale the original image to a small size of 416 pixels × 416 pixels. After feature extraction with Darknet53, the original image will be resized to 13×13 size. Of the three feature maps that have been formed will be combined with two features sized 26 × 26 and 26 × 26 [9]. In each feature map, it predicts three bounding boxes using three anchor boxe and chooses the most appropriate object.

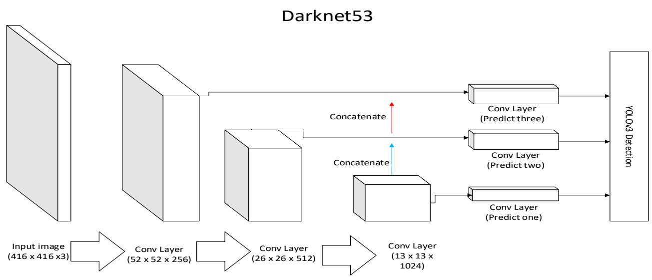


Fig. 1. YOLOv3 network architecture

A. Data Acquisition Stage

As in previous studies that have been carried out, the collection of solar panel surface datasets is carried out with unmanned aircraft tools or often called *Unmanned Aerial vehicles* (UAV) [10]. Data collection is the first process carried in research. Data collection was carried out at one Solar Power Plant on Karanrang Island. The database image collection is carried out with the following materials and equipment.

- The series of monocrystalline solar panels are connected in series and parallel, with a maximum power of 90 W for each panel.
- *Unmanned camera* (UAV), to assist in aerial imagery of the surface of the solar module and mobile camera.

TABLE II. Solar Panel Specifications.

Number of cells	*Dimension (mm)*	*Open Circuit voltage (V)*	*Short Circuit current (A)*	*Current at max power (Imp)*	*The Voltage at max power (Vmp)*	*Voltage Range (V)*
72	805 mm ×1570 mm ×50 mm	44,1 V	5,52 A	5,06 A	35,6 V	24 V

TABLE III. Tools Spesifications.

Tools	*Hovering time full payload (min)*	*Max speed of Ascent (m/s)*	*Max speed of descent (m/s)*	*Operating temperature (℃)*	*Wide*
Mavic air 2	33	4 m/s	3 m/s	−10℃ *to* 40℃	48 MP
iPad Pro 2020	-	-	-	-	12 MP, *f*/1.8

The picture was taken on a sunny day from 08:00 to 10:00 and 16:00 to 17:00 WITA. The UAV is flown and positioned as high as 2 m from the distance of the solar panels as shown in Fig. 2.

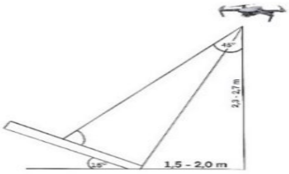


Fig. 2. UAV position when shooting the surface of the solar module.

B. *The proposed method*

The development of the Convolutional Neural Network method with the YOLOv3 method algorithm makes the object detection process a single regression, which processes directly from the pixels of the image that has been collected and creates manually the bounding box coordinates and probability classes. Using YOLO, the system will only see once (You Only Look Once) on the collected image to predict the target or object we have specified [6]. The workflow can be seen in Fig. 3.

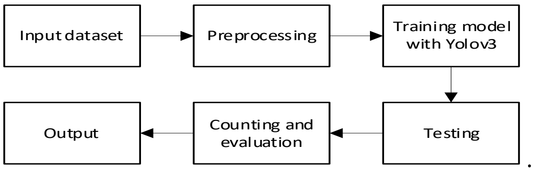


Fig. 3. The proposed system.

1) Pre-Processing

We can pre-process the data before the training and testing process. The pre-processing stage serves for the training process consisting of two stages, namely the process of annotating images, resizing, cropping data. Fig. 4 shows the process of pre-processing data.

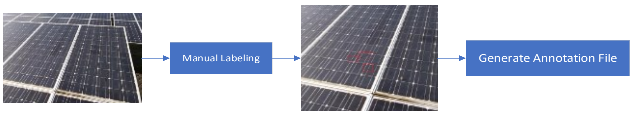


Fig. 4. Pre-Processing process.

The resizing process is carried out for the purposes of input data, the image is equalized to 416×416 in accordance with the provisions of the reference literature [6]. Image resizing aims to change an image size to the desired size usually to facilitate the process of training data with a smaller image resolution size. YOLO requires an input image with a resolution of 416×416 because this resolution is the optimal resolution for detecting objects, even for detecting small objects. This resolution value is also used to anticipate errors when training with large amounts of image data and varying resolutions. Therefore, every image that is entered into the YOLO neural network will be resized to 416×416 pixels resolution. Image annotation is the process of creating labels by giving border boxes and object names to the image [11]. As in previous versions of YOLO9000 the YOLOv3 system will predict the bounding box that has been created manually by using 4 dimensional clusters as anchor boxes. This cluster will predict each coordinate of the bounding box. One bounding box consists of top and bottom $(C_x\ dan\ C_y)$ and the other bounding box consists of width and height $(P_w\ dan\ P_h)$ [7]. then to produce the predicted length and width according to the following formula.

$$b_x = \sigma(t_x) + C_x \tag{1}$$

$$b_y = \sigma(t_y) + C_y \tag{2}$$

$$b_w = p_w e^{tw} \tag{3}$$

$$b_h = p_h e^{th} \tag{4}$$

During the training process, the system will use the calculation of the number of squared errors. If the real condition for the truth box for some coordinate prediction is assumed to be $t*$, then the calculated basic truth value of the truth box is subtracted by the predetermined prediction [7].

$$\hat{t}* - t* \tag{5}$$

The result of image annotation is a data that contains coordinate information on the location of the label's fibrous border box in the form of text.

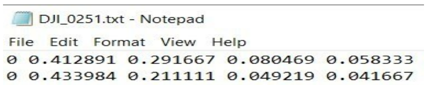

DJI_0251.txt - Notepad
File Edit Format View Help
0 0.412891 0.291667 0.080469 0.058333
0 0.433984 0.211111 0.049219 0.041667

Fig. 5. Image annotation results.

After the dataset to be inputted is collected, and the annotation process has been carried out, the dataset will be obtained which will continue the training process. Data breakdown consists of datasets.

- Datasets are images of objects of the physical condition of the surface of solar panels.
- The collected datasets are about 771 datasets, consisting of 617 training data to be determined for weight and also 155 test data to test the accuracy of the system.

2) Training Stage

Object recognition training is conducted using datasets that have been labeled from image annotations. Then in the training process, the features that have been extracted from the training data using the Convolutional Neural Network layer will be used as input into the Fully Connected Layer which will produce a weight file that will be used for snail trails detection on the surface of the solar panel. The weight file generated from the training process will then be used for the system testing process and used in the counting process. In the convolution process, the model weights are trained so that the system can accurately categorize objects on the dataset. At the training stage, the iteration is determined by an indication of the threshold that has been trained, which is in the form of the value of the error function, as seen in the figure below, the lower the error value, the better the model that detects the object.

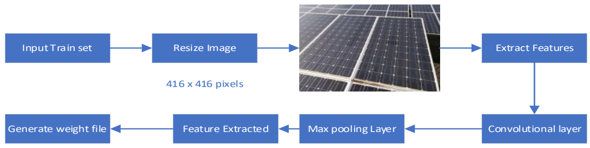


Fig. 6. Training stages process.

In YOLOv3, darknet53 convolution networks are feature extractors, which are used for the training process. Darknet53 consists of several convolution layers that function to combine two sets at dimensions 1×1 and 3×3, with a total of 53 layers. Each convolution layer consists of a batch base layer and a LeakyReLU layer which will give constant values [12]. A number of remaining network modules will be operated in Darknet53 as a form of modification derived from ResNet show in Table 7 [13].

TABLE IV. RESIDUAL MODULE DARKNET-53NETWORK.

	Type	*Filters*	*Size*	*Output*
	Conv	32	3×3	256×256
	Conv	64	3×3/2	128×128
1×	Conv	32	1×1	
	Conv	64	3×3	
	Residual			128×128
2×	Conv	128	3×3/2	
	Conv	64	1×1	
	Conv	128	3×3	
	Residual			64×64
	Conv	256	3×3/2	32×32
	Conv	128	1 × 1	
8×	Conv	256	3×3	
	Residual			32×32
	Conv	512	3×3/2	16×16
	Conv	256	1×1	
	Conv	512	3×3	
8×	Residual		26×26	16×16
	Conv	1024	3×3/2	8×8
	Conv	512	1×1	
4×	Conv	1024	3×3	
	Residual			8×8

Convolutional Neural Network process, every incoming image will go through several stages of the convolutional layer process, namely, filter, polling layer, and fully connected layer. The convolutional layer is the first layer that functions to abstract features to maintain the relationship between pixels in the image by using the matrix box as input data In the extraction feature process using mathematical operations, namely the filter matrix or kernel. Fig. 7 is an illustration of the convolutional process.

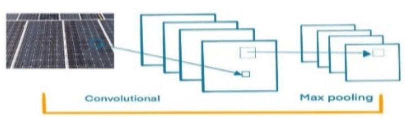


Fig. 7. Feature extraction process.

In the feature extraction process, the filter will shift to the image matrix according to the number of stride. Stride is a filter parameter defined to determine the number of pixel shifts in an image. If the stride value is 1 then the filter will shift every 1 pixel at the time [14]. In the created system there is a pooling layer that serves to reduce the number of parameters used when the image is too large, this is necessary to speed up the computational process of the model. The system used is a type of pooling type Max Pooling which is useful for reducing the dimensions of each map without losing important information [15]. The matrix resulting from the Max pooling layer will be reshaped into a vector shape which is then inserted into a Fully Connected Layer. In this stage the model will be classified to recognize whether the object contained in the image is a snail trail or not, this type of classification is Binary Classification [16].

3) Testing Stage

Fig. 8 is the stage of the testing process, the image dataset has been separated and has gone through the labeling process. The image is collected by drawing a box called a bounding box manually, and it will automatically generate an annotation file containing the coordinates of the middle point of the bounding box, and the height and width of the bounding box will be processed. be input data. In the training process, the features that have been extracted from the training data using the Convolutional Neural Network layer will be used as input into the Fully Connected Layer which will produce a weight file that will be used for the detection of snail trails. In the system made there is a Pooling layer which serves to reduce the number of parameters used when the image is too large, this is needed to speed up the computational process of the model. In the system used is the type of pooling type Max Pooling which is useful for reducing the dimensions of each map without losing important information. The resulting matrix from the Max pooling layer will be reshaped into a vector form which is then inserted into the Fully Connected Layer. In this stage the model will be classified to identify whether the objects in the image are snail trails or not, this type of classification is a Binary Classification. At the detection stage, the object identified by snail trails will immediately form a bounding box that uses a confidence level approach and a class probability map to identify objects.

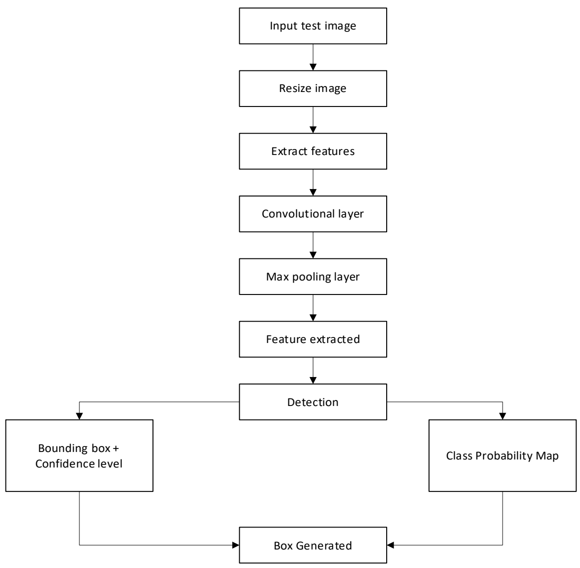


Fig. 8. Block diagram of dataset testing.

III. RESULT AND DISCUSSION

The proposed method will be tested and evaluated with 5 photovoltaic images that have been collected. Image segmentation and evaluation are processed in Google Colab computing. The evaluation process is carried out by calculating accuracy, precision, recall as formulated as follows [17].

$$Accuracy = \left(\frac{TP+TN}{TP+FP+FN\_TN}\right) \times 100\% \quad (6)$$

Performance analysis of snail trail detection systems on solar panels can be done by calculating the accuracy value using the confusion matrix. The accuracy value describes the accuracy of the system in correctly classifying or predicting the data of objects. In other words, the accuracy value is a comparison between the correct object data and the overall data. the confusion matrix used can be seen in the Table 5.

TABLE V. RESIDUAL MODULE DARKNET-53NETWORK.

Class	*Clarified Positive*	*Clarified Negative*
Positive	TP (True Positive)	FP (False Positive)
Negative	FN (False Negative)	TN (True Negative)

The description of the Confusion Matrix is as follows

- *True Positive* is a positive data that snail trails that is classified as true by the system.
- *False Positive* is the sum of positive data from the surface of a solar panel that the system classifies as snail trails.
- *False Negative* is the amount of negative data from snail trail objects that are incorrectly classified by the system.
- *True Negative* is the amount of negative data from the surface of a solar panel that is classified as true by the system not having snail trails.

The initial training was carried out with a max batches value of 1000, then added another 1000 in order to evaluate the accuracy value of the resulting weight file. The figure below is an accuracy diagram resulting from each iteration.

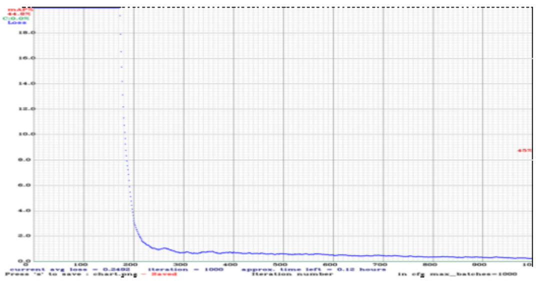


Fig. 9. Mean average precision dataset iteration 1000.

In Fig. 9 the training process is carried out with max batches of 1000, but the results of the accuracy and stability of the training process have not received maximum results, so the results of training with max batches of 1000 cannot be used as a reference for the testing process, because the training process does not recognize images with snail trails objects. on the surface of the solar panel.

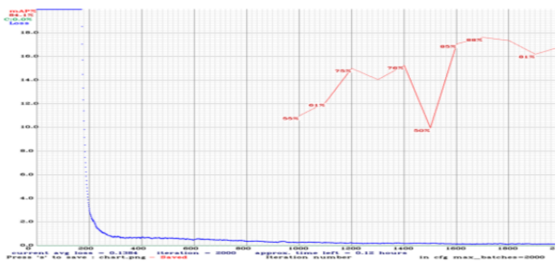


Fig. 10. Mean average precision dataset iteration 2000.

In Fig. 10 the training process is carried out by adding max batches with multiples of 1000 to 2000, the accuracy and stability of the training process has reached a process accuracy of up to 81%. This process has obtained a fairly high accuracy value. This means that the system can function to recognize images with objects that have snail trails or the surface of solar panels that do not have snail trails.

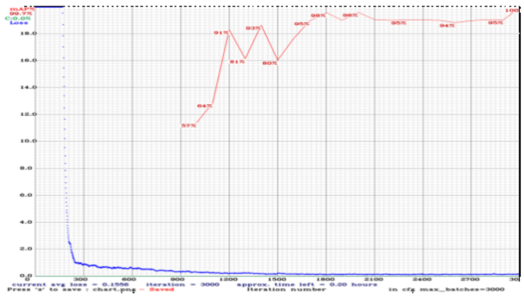


Fig. 11. Mean average precision dataset iteration 3000.

In Fig. 11 the training process is carried out again to get the highest accuracy by adding max batches with multiples of 1000 to 3000, the accuracy and stability of the training process has reached the maximum accuracy of 99.7%. This means that the system can function to recognize and distinguish images with objects that have snail trails or the surface of solar panels that do not have snail trails maximally.

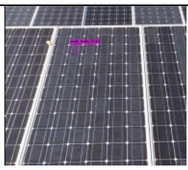
Table 6 shows the results of the system training with a max batches value of 3000 which is the best accuracy, which is 99.7% then from the results of the training the highest data will be tested with a test dataset that has been prepared as much as 20%.

TABLE VI. EVALUATION OF SYSTEM MAP USING CONFUSION MATRIX.

Max Batches	*mAP*	*True Positive*	*False Negative*	*False Positive*	*True Negative*
1000	78,44%	39	5	12	0
2000	80,88%	36	6	9	0
3000	99,7%	32	0	4	0

The system with a max batches value of 3000 is the best accuracy of 99.72%, so the system with a max batches value of 3000 is then tested on a test dataset that has a different resolution size to detect the surface of the solar panel that has snail trails, and no. The results of the System Test can be seen in Table 7.

TABLE VII. EXPERIMENTAL RESULTS.

Dataset	***Dataset testing***	***Object detection testing***	***Results***
DJI_0293			the system detects only 1 out of 2 snail trail object on the solar panel
IMG_1355			the system does not detect the snail trail object on the solar panel
DJI_0262			the system detects 1 snail trail object on the solar panel
DJI_0262			the system detects 2 snail trail object on the solar panel
DJI_0253			the system detects 3 snail trail object on the solar panel

In Table 7, showing the results of the sample testing process for some image data, it can be seen that by using the maximum training data results, the training system can recognize or read snail trail objects precisely and accurately. In our test, randomly modifying images that are in good condition and exposed to snail trails, the results shown for IMG_355 the system can recognize that in the dataset there is no snail trail on the surface. Then we retested the system with a dataset that contained snail trails, the results proved that the system can recognize snail trails and accurately determine the number of snail trails, which are found on the surface of solar cells.

IV. CONCLUSION AND FUTURE WORK

This paper aims to detect snail trail areas that occur in photovoltaic modules. This study shows a deep learning method with the YOLOv3 algorithm to detect objects. images are captured using the camera and collected and then manually labeled resulting in a dataset. Next, the dataset will be trained to generate a weights file. The method used to extract features and classifications is the *Convolutional Neural Network* in which there is a layer. The result of the object detection will be visualized describing the bounding box of the detected object. Bounding boxes will then be calculated to find the system accuracy percentage.

The system produces an average value of 99.7% accuracy. This shows that this work is successful in detecting snail trail objects on the surface of solar panels which can help in checking damage or early prevention in supporting the continued use of solar cells. In addition to the benefits of this study, there are shortcomings that require improvement in further research in the future to develop a damage monitoring system on solar panels.

The framework used in this system is the darknet, but there are still other frameworks such as TensorFlow or Dark flow (a combination of darknet and TensorFlow) that can be used to facilitate the processing of the system.

REFERENCES

[1] G. Perveen, M. Rizwan, N. Goel, and P. Anand, “Artificial neural network models for global solar energy and photovoltaic power forecasting over India,” Energy Sources, Part A: Recovery, Utilization and Environmental Effects, 2020, doi: 10.1080/15567036.2020.1826017.

[2] D. S. Pillai and N. Rajasekar, “A comprehensive review on protection challenges and fault diagnosis in PV systems,” Renewable and Sustainable Energy Reviews, vol. 91. Elsevier Ltd, pp. 18–40, Aug. 01, 2018. doi: 10.1016/j.rser.2018.03.082.

[3] S. Meyer et al., “Silver nanoparticles cause snail trails in photovoltaic modules,” Solar Energy Materials and Solar Cells, vol. 121, pp. 171–175, 2014, doi: https://doi.org/10.1016/j.solmat.2013.11.013.

[4] H.-C. Liu, C.-T. Huang, W.-K. Lee, S.-S. Yan, and F.-M. Lin, “A Defect Formation as Snail Trails in Photovoltaic Modules,” Energy Power Eng, vol. 07, no. 08, pp. 348–353, 2015, doi: 10.4236/epe.2015.78032.

[5] IEEE Staff, 2018 Chinese Automation Congress (CAC). IEEE, 2018.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[7] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” Apr. 2018, [Online]. Available: http://arxiv.org/abs/1804.02767

[8] L. Zhao and S. Li, “Object detection algorithm based on improved YOLOv3,” Electronics (Switzerland), vol. 9, no. 3, Mar. 2020, doi: 10.3390/electronics9030537.

[9] H. Ma, Y. Liu, Y. Ren, and J. Yu, “Detection of collapsed buildings in post-earthquake remote sensing images based on the improved YOLOv3,” Remote Sens (Basel), vol. 12, no. 1, Jan. 2020, doi: 10.3390/RS12010044.

[10] E. Alfaro-Mejía, H. Loaiza-Correa, E. Franco-Mejía, A. D. Restrepo-Girón, and S. E. Nope-Rodríguez, “Dataset for recognition of snail trails and hot spot failures in monocrystalline Si solar panels,” Data Brief, vol. 26, Oct. 2019, doi: 10.1016/j.dib.2019.104441.

[11] J. Redmon and A. Farhadi, “YOLO9000: Better, Faster, Stronger.” [Online]. Available: http://pjreddie.com/yolo9000/

[12] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” in Proceedings of the 32nd International Conference on Machine Learning, Aug. 2015, vol. 37, pp. 448–456. [Online]. Available: https://proceedings.mlr.press/v37/ioffe15.html

[13] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition.” [Online]. Available: http://image-net.org/challenges/LSVRC/2015/

[14] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva, “Land Use Classification in Remote Sensing Images by Convolutional Neural Networks,” Aug. 2015, [Online]. Available: http://arxiv.org/abs/1508.00092

[15] M. B. Bejiga, A. Zeggada, A. Nouffidj, and F. Melgani, “A convolutional neural network approach for assisting avalanche search and rescue operations with UAV imagery,” Remote Sens (Basel), vol. 9, no. 2, 2017, doi: 10.3390/rs9020100.

[16] S. Albelwi and A. Mahmood, “A framework for designing the architectures of deep Convolutional Neural Networks,” Entropy, vol. 19, no. 6, Jun. 2017, doi: 10.3390/e19060242.

[17] O. Caelen, “A Bayesian interpretation of the confusion matrix,” Ann Math Artif Intell, vol. 81, no. 3, pp. 429–450, 2017, doi: 10.1007/s10472-017-9564-8.