

Detection of microcracks and dark spots in monocrystalline PERC cells using photoluminescence imaging and YOLO-based CNN with spatial pyramid pooling

Amran Binomairah¹, Azizi Abdullah², Bee Ee Khoo¹, Zeinab Mahdavi-pour³, Teow Wee Teo³, Nor Shahirah Mohd Noor¹, and Mohd Zaid Abdullah^{1,*}

¹ School of Electrical and Electronics Engineering, Engineering Campus, Universiti Sains Malaysia, 14300 Nibong Tebal, Penang, Malaysia

² Faculty of Information Sciences and Technology, Universiti Kebangsaan Malaysia, 43600 Bangi Selangor, Malaysia

³ TT Vision Technologies Sdn. Bhd., Plot 106, Hilir Sungai Keluang 5, Bayan Lepas Industrial Zone, Phase 4, Penang 11900, Malaysia

Received: 21 July 2022 / Received in final form: 11 October 2022 / Accepted: 2 November 2022

Abstract. Two common defects encountered during manufacturing of crystalline silicon solar cells are microcrack and dark spot or dark region. The microcrack in particular is a major threat to module performance since it is responsible for most PV failures and other types of damage in the field. On the other hand, dark region in which one cell or part of the cell appears darker under UV illumination is mainly responsible for PV reduced efficiency, and eventually lost of performance. Therefore, one key challenge for solar cell manufacturers is to remove defective cells from further processing. Recently, few researchers have investigated deep learning as an alternative approach for defect detection in solar cell manufacturing. The results are quite encouraging. This paper evaluates the convolutional neural network based on heavy-weighted You Only Look Once (YOLO) version 4 or YOLOv4 and the tiny version of this algorithm referred here as Tiny-YOLOv4. Experimental results suggest that the multi-class YOLOv4 is the best model in term of mean average precision (mAP) and prediction time, averaging at 98.8% and 62.9 ms respectively. Meanwhile an improved Tiny-YOLOv4 with Spatial Pyramid Pooling scheme resulted in mAP of 91.0% and runtime of 28.2 ms. Even though the tiny-weighted YOLOv4 performs slightly lower compared to its heavy-weighted counterpart, however the runtime of the former is 2.2 order much faster than the later.

Keywords: Solar cell / microcrack / dark region / CNN / YOLO

1 Introduction

Solar energy has gained increasing attention as a source of renewable energy in recent years as the price of oil has increased and environmental concerns have grown. However, the efficiency of the system is reduced due to the solar cell defects that occur during production or operation. Among many types of defects the microcrack and dark region are high in the list of worries for the PV industry. On average, about 5–10% of fully completed solar cells coming out of a production line contains these defects, in particular microcrack. Eliminating these defects will aid in increasing the efficiency of PV generation and the lifetime of solar cells. However, detecting these defects can be challenging due to their variety in shape and size.

Furthermore, these defects which are located below the surface of solar cells may be invisible in the image captured by a standard CCD camera. Hence, specialized imaging systems have to be deployed in order to inspect PV cells for defects. To-date, various imaging technologies such as the electroluminescence (EL) [1,2], The infrared (IR) [3] and the photoluminescence (PL) [4] have been developed for solar cell and wafer inspection. A study which reviewed existing and emerging imaging technologies used for solar wafer or cell inspection concluded that PL remains popular among machine vision manufactures due to its speed and cost advantages [5]. Interested readers who wish to learn more about these technologies are referred to this publication. On software inspection, several researchers have developed different techniques and approaches for inspecting solar cells and PV modules automatically. These approaches can be divided into two categories. They are (i) the image processing technique and (ii) the deep

* e-mail: mza@usm.my

learning method. This study investigates a deep learning approach based on YOLOv4 algorithm for training a model capable of detecting microcrack and dark region in solar cell images captured using PL system. Both the heavy-weighted and light-weighted YOLOv4 networks have been used in the investigation from which an improved Tiny-YOLOv4 has been proposed.

2 Related works

2.1 Defect detection based on image processing approach

Traditional image processing approaches have been widely used to identify defects in solar cells and modules such as microcracks and finger interruptions. Most of these techniques are based on color, shape, and texture of defects. An algorithm based on anisotropic diffusion filtering was proposed in [6] to detect small visible cracks on the surface of solar wafers. The image captured with a standard CCD camera and light source reveals microcracks with low grey scales and strong gradients. Diffusion preserves the crystal grain background while smoothing out the suspected defect region. By subtracting the diffused image from the original image, the location of the crack can be determined precisely. Meanwhile [7] proposed an anisotropic diffusion filter and image segmentation method for identifying microcracks in the presence of noise. A method based on spectral clustering to cluster the features from the regions of interest is proposed. The method achieves high efficiency when detecting finger interruptions in polycrystalline cells [8]. Chen et al. [9] proposed a detection method that can precisely locate cell cracking in EL images using steerable evidence filtering. Although these studies are very promising, however, most authors considered microcrack defect of finger interruption only. To address this drawback, a method based on Fourier image reconstruction has been proposed in [10] that successfully detects breaks, finger interruptions, and microcracks. Even though, this method is proven effective, however, the algorithm is very time consuming, and requires high end workstation in order to deliver a real-time solution.

2.2 Defect detection based on deep learning approach

Few researchers have successfully applied deep learning approach to inspect PV cells in industrial set-ups. Both polycrystalline and monocrystalline cells have been considered. For instance, two automatic CNN defect detection techniques based on improved VGG-19 and SVM algorithm have been proposed [11]. By rounding-up the continuous probability prediction to the nearest neighbor of the four original classes enabled these authors to compare the precision of the algorithm with the ground truth label directly. The results indicate that CNN is more accurate than SVM methods. In another study Li et al. [12] proposed a highly accurate diagnostic model for defects detection of PV modules in a large-scale solar farm using unmanned aerial vehicles and CNN. The results are very encouraging with accuracy reaching more than 90% in controlled environment. Similarly, [13] proposed an automatic defect classification method based on CNN model that achieved a higher

classification accuracy than existing methods. Although the aforementioned studies achieved competitive accuracies, however, the methods and procedures are very computationally intensive, making these techniques highly dependent on expensive and costly hardware. Addressing this problem led to the development of lightweight CNN network architecture which is referred in the literatures as the Faster-RCNN [14]. Results indicate that this architecture yielded satisfactory results with few calculations. A further improvement utilizing the genetic algorithm for feature extraction has been reported following the publication of GA-Faster-RCNN technique [15]. The improved technique can identify cell defects and mark their locations automatically. In another research, Zhang et al. [16] designed a detection algorithm that combines results from Faster R-CNN and R-FCN in order to improve detection precision and position accuracy. The authors reported achievement of 85.7% in terms of the mean average precision (mAP) which is low by industrial standard. A much more complex deep learning architecture has also been developed by combining there different CNN technologies – (i) the Faster-RNN, (ii) the EfficientNet, (iii) the autoencoder [17]. Such a combination enabled the authors to construct an end-to-end deep learning pipeline that detects, localizes, and segments cell-level anomalies from the entire photovoltaic modules. This new architecture has also improved the sensitivity of Faster-RNN when detecting defects in solar cells. In this case the autoencoder not only allowed the anomaly in PV modules be segmented, but it also facilitated defect detection. In another work a CNN based on generative adversarial network has been evaluated for detecting anomalies in solar cells [18]. Although tested on different datasets, such a network performs less satisfactorily with recall, precision and specificity averaging at 79%, 73% and 73% respectively. More recently an empirical digital twin which fuses measurement data and expert knowledge has been utilized for detecting certain types of defects in solar wafers [19]. Though promising, however, this method requires *IV* measurements which is difficult to be implemented in high-speed production settings. From the brief literature review, currently, there are three main issues associated with the application of deep learning for defect detection involving solar cells or PV modules. Briefly they are (i) low detection accuracy, (ii) slow detection speed, and (iii) lack of standard for defect detection. Although existing methods have significantly reduced inspection time, however, they are incapable of processing images in real time. In some environments with sensitive hardware resources and stringent real-time requirements, its application remains difficult. In term of business and economy, high over and under rejection due to poor recall and precision respectively, would impact the industry negatively. Hence, additional research is required.

3 YOLOV4 algorithm description

You Only Look Once, or more commonly referred to as YOLO, is a solution for object detection comprising of a single neural network [20]. In operation YOLO takes images or video frames as input and generates bounding

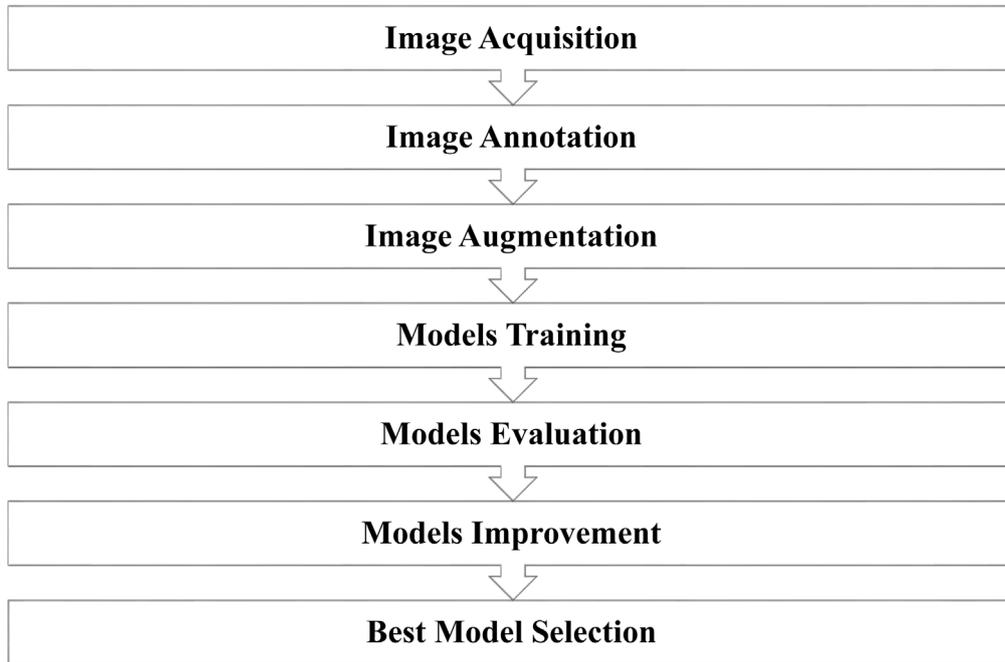


Fig. 1. Flowchart summarizing the design of deep learning models.

boxes and the probability of a class being within these boxes. A single convolutional network predicts multiple bounding boxes simultaneously, requiring only a single look at an image to determine which objects are present and where they are. Meanwhile YOLOv4 is an enhanced version of the original YOLO architecture which was introduced in 2020. Since then YOLOv4 is very popular among the AI community because of its accuracy and detection precision. More importantly, this algorithm, in particular the Tiny-YOLOv4 is architecturally less complicated compared to other types of convolution neural networks, making this algorithm suitable for high-speed applications such as the PV production. The backbone, network training, activation function, and loss function of YOLOv4 are optimized, resulting in a faster operation without compromising the accuracy and sensitivity of the algorithm [21]. To train and extract image features, YOLOv4 uses CSPDarknet53 as the main backbone engine. The latter is an open-source neural network which is freely available online. Meanwhile the Path Aggregation Network or PANet [22] is used as feature extractor while object detection is achieved by means of standard YOLOv3 algorithm [20].

Tiny-YOLO, on the other hand, is a derivative of YOLO with the former containing fewer convolution layer in the backbone [22]. Hence Tiny-YOLO is essentially YOLO but with reduced size. For this reason, Tiny-YOLO can perform real-time detection on devices with low computing power and has a faster detection speed while maintaining the overall network accuracy. This architecture makes use of the CSPdarknet53-Tiny backbone network and the FPN network for enhanced feature extraction. For prediction this algorithm employs filters of sizes 13×13 and 26×26 in first and second layers

respectively. Meanwhile the LeakyReLU function is used to activate the entire network. Prior to detection, each input image is scaled to 416×416 and 608×608 size pixels. Both the CBL and Resblock modules serve as the network's backbone. The former comprises of the convolution layers, the batch normalization block and the LeakyReLU activation function while the later consists of a residual network whose structure is similar to CSPNet [22]. The algorithm, specifically the feature extraction network, employs a pyramid structure to improve the network's feature fusion and detection precision. Like other deep learning frameworks, YOLOv4 or Tiny-YOLOv4 needs to be fine-tuned in order to ensure acceptable performance in term of accuracy and speed. For this reason, a number of YOLOv4 models are trained in this study using both heavy-weighted and light-weighted models. Further improvement of the algorithm is achieved by incorporating the Spatial Pyramid Pooling (SPP) in the neck part of the network's backbone. The details are discussed in the following sections.

3.1 Methodology

Overall, the methodology in designing training models is summarized in Figure 1. The process starts with image acquisition, then followed by image annotation and augmentation. Models training constitutes the next subsequent stage. This is immediately followed by fine-tuning and evaluating each model using validation and test samples respectively. Based on results from previous step, few models are selected for further improvement after which a best model is selected for deployment. The later constitutes the last or final step.

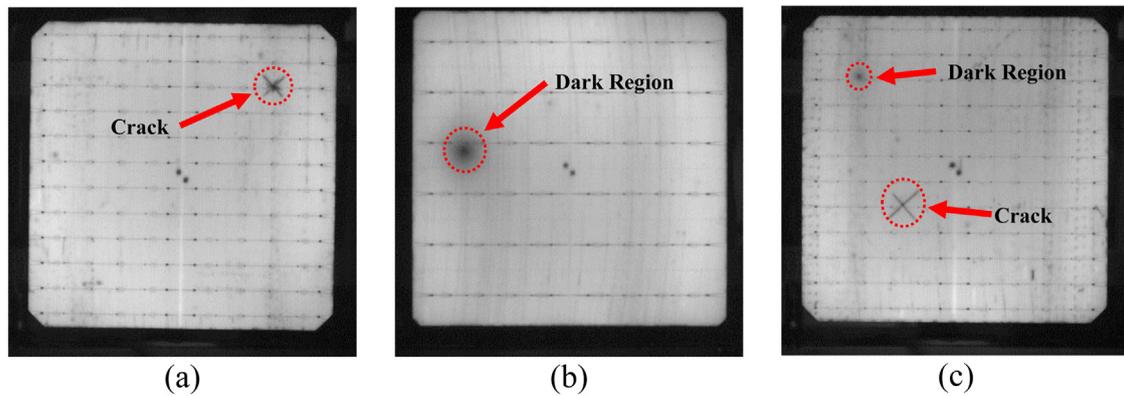


Fig. 2. An example of PL image from each dataset (a) Dataset 1, (b) Dataset 2, and (c) Dataset 3.

3.1.1 Image acquisition

Acquiring images of solar cells is a critical step in the training. In this study the PL technique is used to acquire solar cells images. Examples of PL images containing dark region and microcrack defects are shown in Figure 2. The PL set-up used in capturing this image comprises mainly of an excitation source, camera, optical lens, and long pass filter. In this case a monochromatic light with wavelengths ranging from 620 to 645 nm is used as an excitation source. Meanwhile the ICX285AL CCD camera manufactured by Sony Corporation together with SWIRON 1.4/23 optical lens manufactured by Schneider are used as the main image capturing device. This camera with resolution of 1434×1050 pixels is capable of delivering high quality image with sensitivity that extends into the NIR region. A 1000 nm long pass filter manufactured by Edmund Optics Inc. is used to block an excitation light in the visible range while permitting the transmission of PL signals only. Including the exposure time, on average it takes less than 1s to capture one solar cell with standard size of 125×125 mm or 156×156 mm. All solar cells used in this study are manufactured using the Passivated Emitter Real Cell (PERC) technology. In this work both single and multi-class models are used to train the network. Hence three different datasets are created in order to evaluate these models. Herein these datasets are referred to as Dataset 1, Dataset 2 and Dataset 3. Dataset 1 which contains 160 microcrack samples is used to train and develop single class microcrack model. Meanwhile the dark region model is developed using Dataset 2 which comprises of 220 solar cells images with dark region artefact. Dataset 3 which contains 501 solar cells images are used for training and developing the multi-class model. A training-to-test sample ratio of approximately 7:3 is applied to all three datasets. This ratio is reasonably adequate for most deep learning applications including solar cell inspection discussed in this paper.

3.1.2 Image annotation

All images captured in the datasets have been inspected and verified by expert human inspectors. This helps ensuring correct labeling and minimizes uncertainty in the data. The software tool LabelImg is used to label all defects

in solar cell images during training, resulting in a text file for each image containing the defect's id and the coordinates of the bounding boxes. In the case of Dataset 3, this tool is used to manually label all defective samples into two classes –0 and 1 corresponding to microcrack and dark region class respectively. In this dataset the sizes of microcrack and dark region range from 2 mm to 110 mm and 6 mm^2 to 1309 mm^2 respectively. Meanwhile the labeling is performed separately for Dataset 1 and Dataset 2 and the class id is set to 0 as all samples in each dataset belong to one class only.

3.1.3 Image augmentation

Increasing the diversity of samples can help improve identification accuracy. To increase the richness of the experimental data and the model's generalizability, the data augmentation is used to increase the number of samples in the datasets. Two popular augmentation techniques are used for this purpose. They are (i) flipping and (ii) rotating. Both horizontal and vertical are used in flipping which rotations involved 90° and 270° . Each image in the training sets is processed using these augmentation techniques, and the resulting text file for each augmented image is annotated simultaneously. Altogether 1755 additional images are produced after augmentation resulting in a total of 2106 images for Dataset 3. Similarly, an additional 800 microcrack images and 1100 dark region images are created, yielding a total of 960 and 1320 samples in Dataset 1 and Dataset 2 respectively. The validation set is taken as a ratio of 15% from each training set yielding a total of 144, 198 and 316 samples in Dataset 1, Dataset 2 and Dataset 3 respectively.

3.1.4 Model training

All deep learning models discussed in this paper are implemented on Intel i5-4460 personal computer with 20 G memory, and operating at an optimum speed of 3.2 GHz. This PC is equipped with 8 GB NVIDIA GeForce GTX 1070 GPU and running in Windows 10 operating system.

Altogether fourteen YOLOv4-based deep learning models are designed and trained with different configurations and parameters. In this case Model 1 and Model 2 are based on heavy-weighted YOLOv4 and the remaining models (Model 3–Model 14) are essentially Tiny-YOLOv4

Table 1. Important parameter settings and configurations of various YOLOv4 models.

Model	Network	Input size	Subdivisions	Detection layers	Activation function
Model 1	Original	416×416	32	3	Leaky
Model 2	Original	608×608	64	3	Leaky
Model 3	Tiny	416×416	8	2	Leaky
Model 4	Tiny	416×416	8	2	Mish
Model 5	Tiny	416×416	8	3	Leaky
Model 6	Tiny	416×416	8	3	Mish
Model 7	Tiny	416×416	4	2	Leaky
Model 8	Tiny	416×416	4	2	Mish
Model 9	Tiny	416×416	4	3	Leaky
Model 10	Tiny	416×416	4	3	Mish
Model 11	Tiny	608×608	8	2	Leaky
Model 12	Tiny	608×608	8	2	Mish
Model 13	Tiny	608×608	8	3	Leaky
Model 14	Tiny	608×608	8	3	Mish

algorithm. The parameters considered in the design are the size of the input image, the number of subdivisions, the number of detection layers and the activation function. In the case of activation function, the popular LeakyReLU is tested alternately with the Mish functions while keeping other parameters unchanged. [Table 1](#) summarizes important configurations and parameter settings for all models. The architecture of heavy-weighted YOLOv4 with input size of 608×608 is shown in [Figure 3](#). This architecture forms the basis in designing Model 1 and Model 2. Their differences are in term of number of subdivision and size of input image. In this case Model 1 is designed with 32 subdivisions and input size of 416×416 . Model 2 is slightly large having 64 subdivisions and input size of 608×608 . Smaller image size and subdivisions in Model 1 allow GPU to handle more images compared to Model 2. Meanwhile [Figure 4](#) shows the architecture for Tiny-YOLOv4, with input size of 416×416 , detection layers, Leaky ReLU activation function. This architecture forms the basis of Model 3–Model 14. The tiny models with different parameter settings and types of activation functions are shown in [Table 1](#). Compared to heavy-weighted YOLO, the lighted-weighted YOLO enables training using much smaller subdivisions, and hence allowing GPU to handle

more images simultaneously during training. In this way the runtime is improved drastically as evident from results discussed in the following section. All models discussed here are evaluated using the same samples in the datasets, from which the best model is selected for further improvement.

3.1.5 Model evaluation

The performance of the algorithm is assessed using a popular indicator based on mean average precision or mAP. This indicator is computed by first forming the precision-recall (PR) curve, and then calculating the average precision (AP) for each class by means of integration. The mAP is simply the average of AP over all classes. Mathematically:

$$AP = \int_0^1 P(R) dR \quad (1)$$

and,

$$mAP = \frac{1}{N_c} \sum_{n=1}^{N_c} AP_n \quad (2)$$

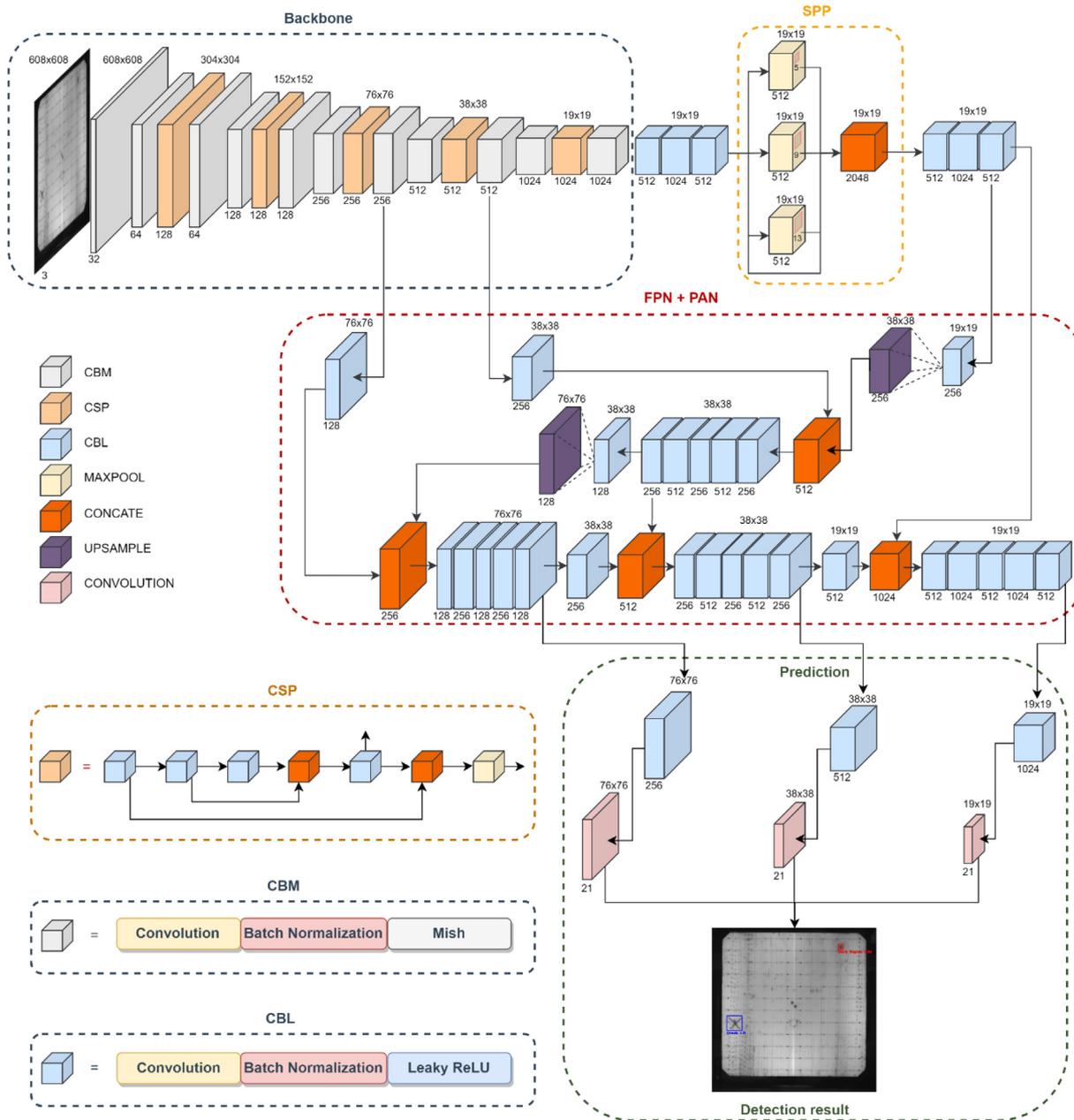


Fig. 3. The architecture of the original heavy-weighted YOLOv4.

where AP_n is the AP for the n th class, and N_c is the total number of classes. In this study N_c is set to 1 and 2 for single and multi-class models respectively.

4 Results and discussions

Each model discussed in the previous section is first evaluated using validation samples and mAP value calculated. The process is repeated for test samples after which the mAP is averaged out. Results comparing the performance of each model are plotted graphically in Figures 5–7, corresponding to microcrack, dark region and multi-class models respectively.

For original heavy-weighted Yolov4, it can be seen from Figures 5–7 that Model 2 performed the best when classifying microcrack defect only, resulting in mAP values of 100% and 98.8% for validation and test samples respectively. This corresponds to an average mAP of approximately 99.9%. In comparison Model 7 is the best Tiny-Yolov4’s model for microcrack detection with mAP values of 99.3% and 99.6% calculated from validation and test samples respectively. This is equivalent to an average mAP of 99.4% which is slightly lower than results from Model 2. Meanwhile the original Yolov4 in Model 1 is the best single model for dark region, with mAP averaging at 93.9% compared to 82.6% mAP for Tiny-Yolov4 in Model 7. Similarly, the original Yolov4 in Model 2 is the best

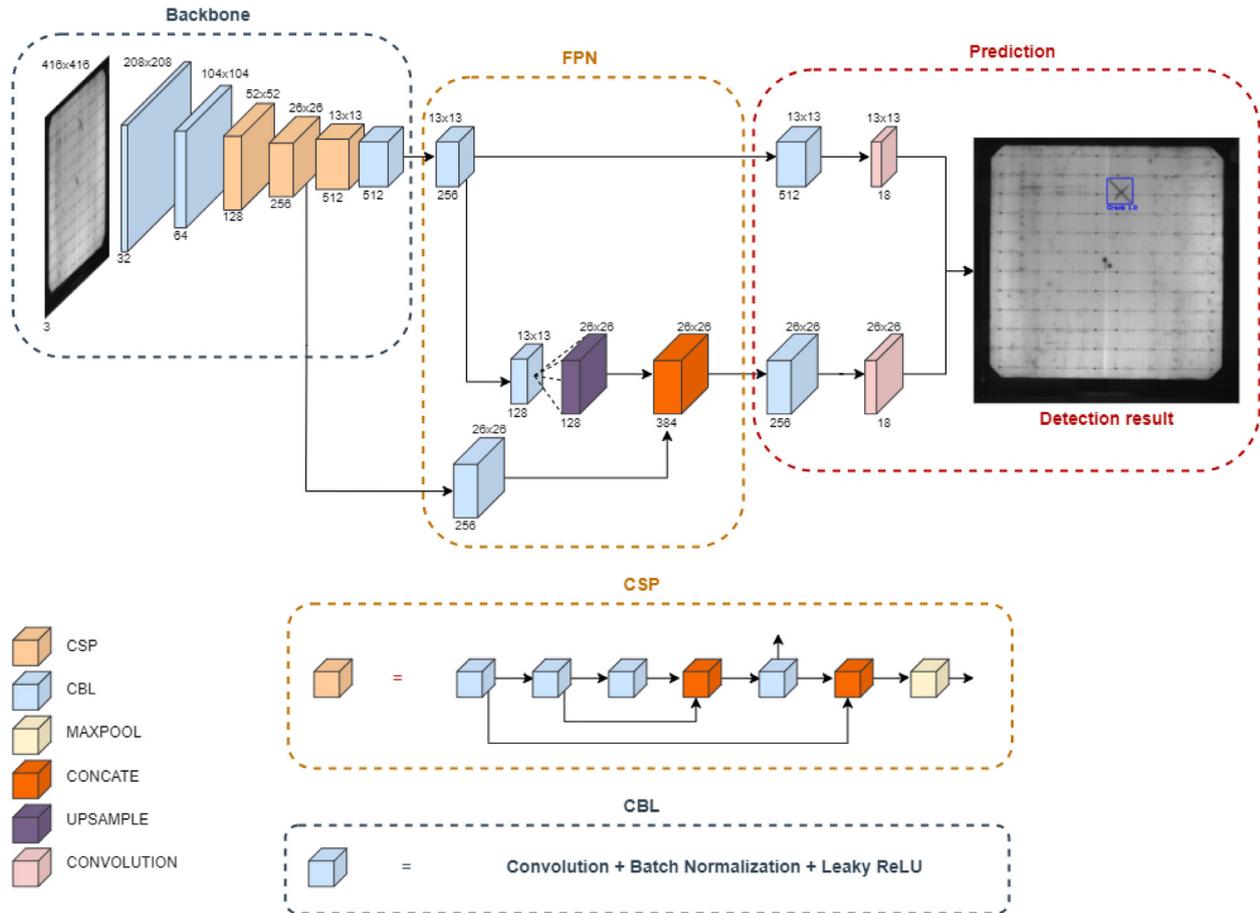


Fig. 4. The architecture of original light-weighted Tiny-YOLOv4.

multi-class performer, resulting in an average mAP of 98.8%. This value is significantly higher compared to Tiny-Yolov4 in Model 14 which produced an average mAP of 90.1%. In terms speed, the original YOLOv4 is the slowest with training and prediction times averaging at approximately 20 h and 63 ms respectively. This model is also a very computationally intensive algorithm since it requires 244 GB memory space in order to work efficiently. In contrast, Tiny-YOLOv4 is significantly much faster with training and prediction times averaging at 3.5 h and 26.8 ms respectively. This algorithm also outperforms the original YOLOv4 in terms of hardware resources, with the former requiring 22 MB of memory only. Table 2 summarizes important features of best models, comparing the original and tiny YOLOv4 algorithms.

Close examination of Table 2 reveals few important points. If a heavy-weighted multi-class model is formed by combining two single class models, i.e. Model 2 micro-crack + Model 1 dark region, then the resulting network would have an accuracy of 96.9% mAP, 107 ms prediction time and 488 MB memory size. Clearly the new model is not as good as Model 2 multi-class since the latter has mAP, prediction time and memory size averaging at 98.8%, 26.3 ms, and 244 MB respectively. Hence a heavy-weighted model trained to perform multi-class detection performs much better than a multi-class model formed by a combination of two heavy-weighted single class models.

In contrast the light-weighted multi-class model formed by a combination of two tiny models, i.e. Model 7 micro-crack + Model 7 dark region, would result in mAP, prediction time and memory size averaging of at 91.0%, 52.7 ms and 44.8 MB respectively. In terms of mAP, clearly, a new tiny model performs slightly better than a same model when trained to perform multi-class detection, i.e. Model 14. Nevertheless the prediction time and memory size of this new model is significantly much higher compared to Model 14. Table 3 summarizes important findings comparing original and combined models for both heavy-weighted and tiny YOLOv4. Referring to this table, evidently, the tiny original and combined models perform slightly lower than their heavy-weighted counterparts. Nevertheless, their prediction times, particularly the original multi-class, is at least twice much faster compared to heavy-weighted YOLOv4.

Like other manufactured products, PV production is usually automated with throughput reaching 3600 samples per second in most cases. Therefore, prediction time is a very important factor to consider. In terms of speed, it's extremely difficult for heavy-weighted YOLOv4 to meet this requirement as the above results suggest. In contrary a light-weighted tiny YOLOv4 offers much more practical solution due to its superiority in speed. In terms of accuracy, however, this network is slightly inferior compared to its heavy-weighted counterpart. Hence, a study has been initiated,

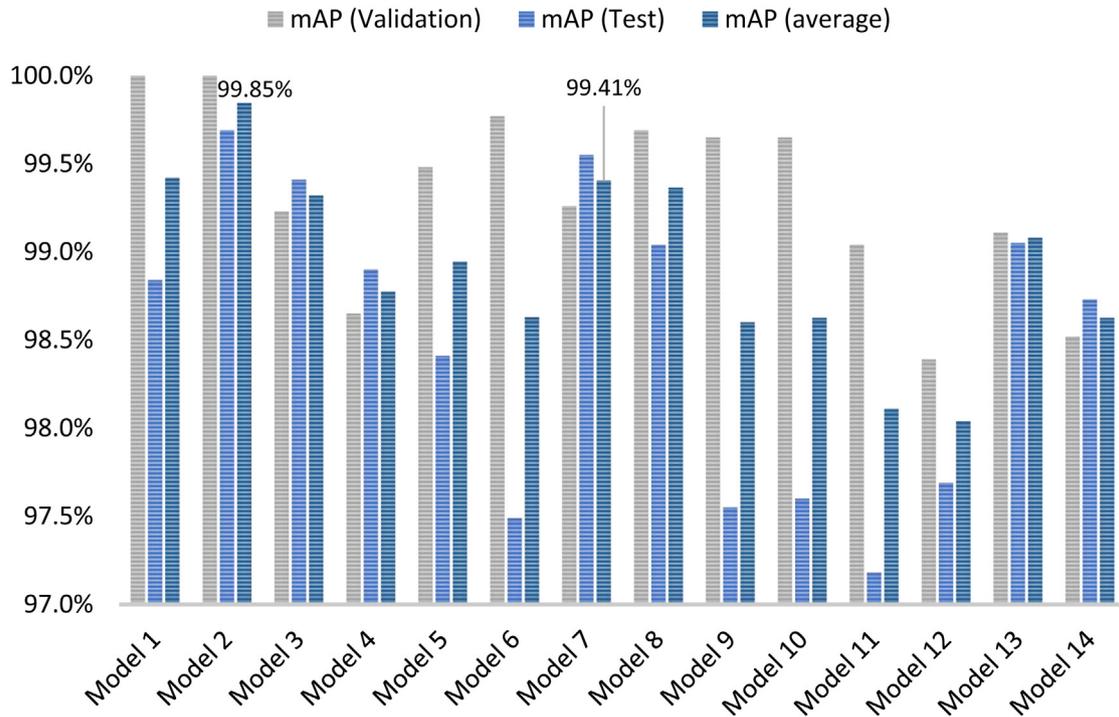


Fig. 5. The mAP values calculated from each model when performing single-class detection (microcrack).

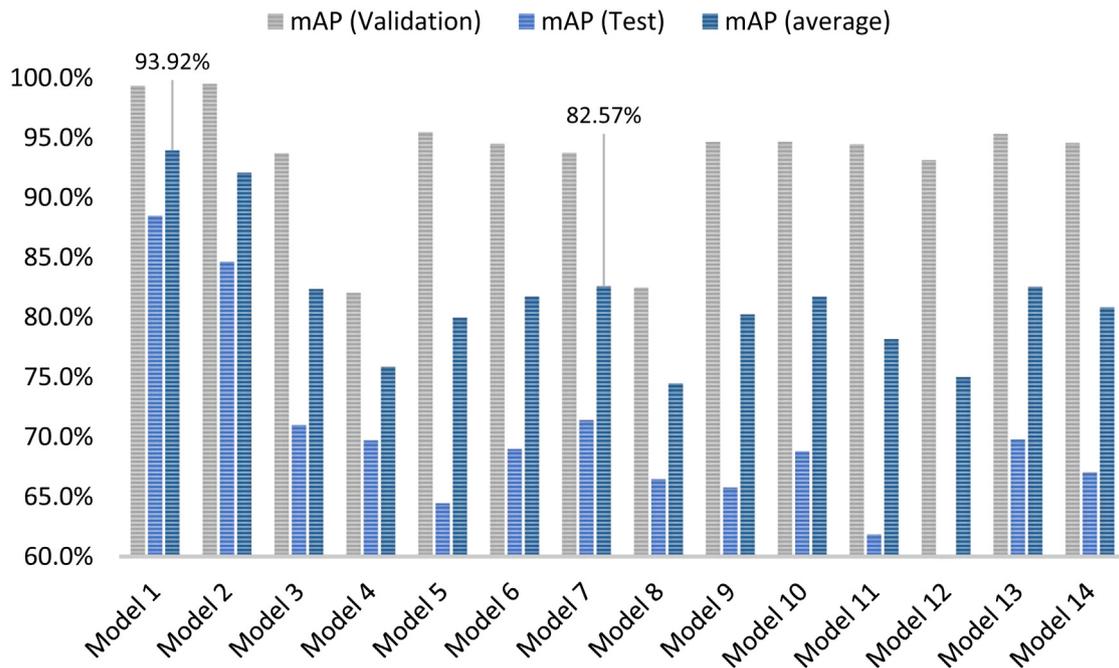


Fig. 6. The mAP values calculated from each model when performing single-class detection (dark region).

aiming to improve the performance of Tiny-YOLOv4. This study is based on assumption that small defects occupy fewer pixels, and hence, taking up smaller area in the receptive field. If these features can be amplified then they can be extracted and used for further processing. The SPP is a good candidate for this task since this network is capable of expanding the receptive field, thus enabling the multi-level

features be extracted more efficiently [23]. Furthermore the SPP concatenates the outputs of the max pooling, leading to further enhancement of the detailed expression of small defects. Consequently, the recognition ability of the algorithm could be improved. In testing this hypothesis, a single class model (Model 7) is modified by adding SPP which is composed of primarily 13×13 filter kernels. The kernel's

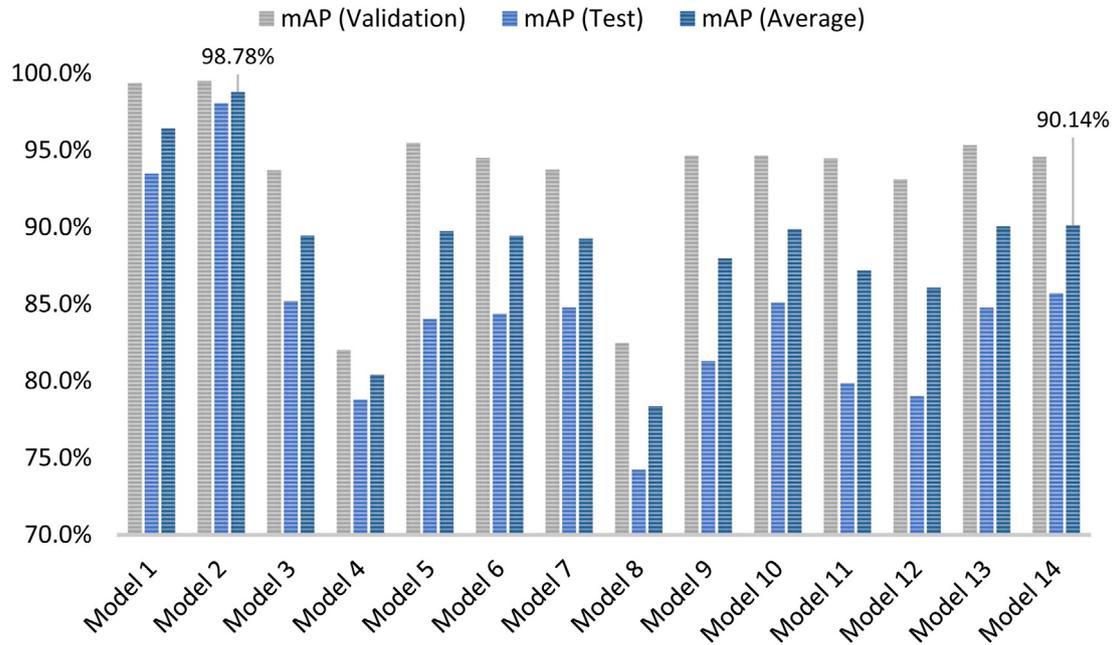


Fig. 7. The mAP values calculated from each model when performing multi-class detection (microcrack and dark region).

Table 2. Best single and multi-class models comparing original and tiny YOLOv4.

Model	Network	Best models	mAP (Validation) (%)	mAP (Test) (%)	mAP (Average) (%)	Prediction time (ms)	Training time (hrs)	Memory size (MB)
Single class (microrack)	Original	Model 2	100.0	98.8	99.9	62.9	26.0	244
	Tiny	Model 7	99.3	99.6	99.4	26.9	2.9	22.4
Single class (dark region)	Original	Model 1	99.4	88.5	93.9	41.8	14.7	244
	Tiny	Model 7	93.7	71.4	82.6	25.8	3.0	22.4
Multi-class (Microcrack and dark region)	Original	Model 2	99.5	98.0	98.8	62.9	26.3	244
	Tiny	Model 14	94.6	85.7	90.1	27.8	5.0	23.3

Table 3. The performance of multi-class heavy-weighted and tiny YOLOv4, comparing original and models formed by a combination of two single class detectors.

Model type	Average mAP (%)	Prediction time (ms)
Combined multi-class heavy-weighted	96.9	104.7
Original multi-class heavy-weighted	98.8	62.9
Combined single class tiny	91.0	52.7
Original single class tiny	90.1	27.8

size matches with input image which has been resized to 416×416 pixels. The resulting model is shown in Figure 8. The same method is used to modify a multi-class model (Model 14), and the result is shown in Figure 9. In this case the kernel size is fixed to 19×19 in order to match with resized image of 608×608 pixels.

The modified architecture are again evaluated using images in Dataset 1, Dataset 2 and Dataset 3, and the results are shown in Table 4.

Referring to Table 4, it can be seen that the performance of the modified network, in particular the multi-class model, has registered a slight improvement

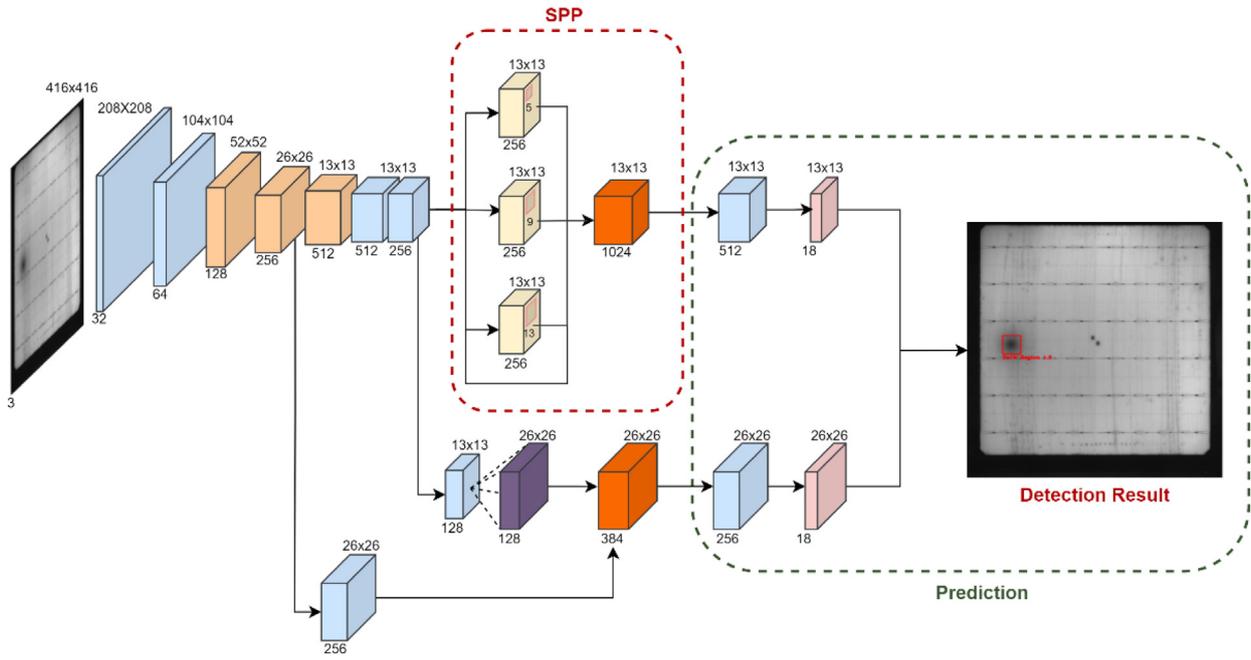


Fig. 8. Modified single class Tiny-YOLOv4 with SPP extractor.

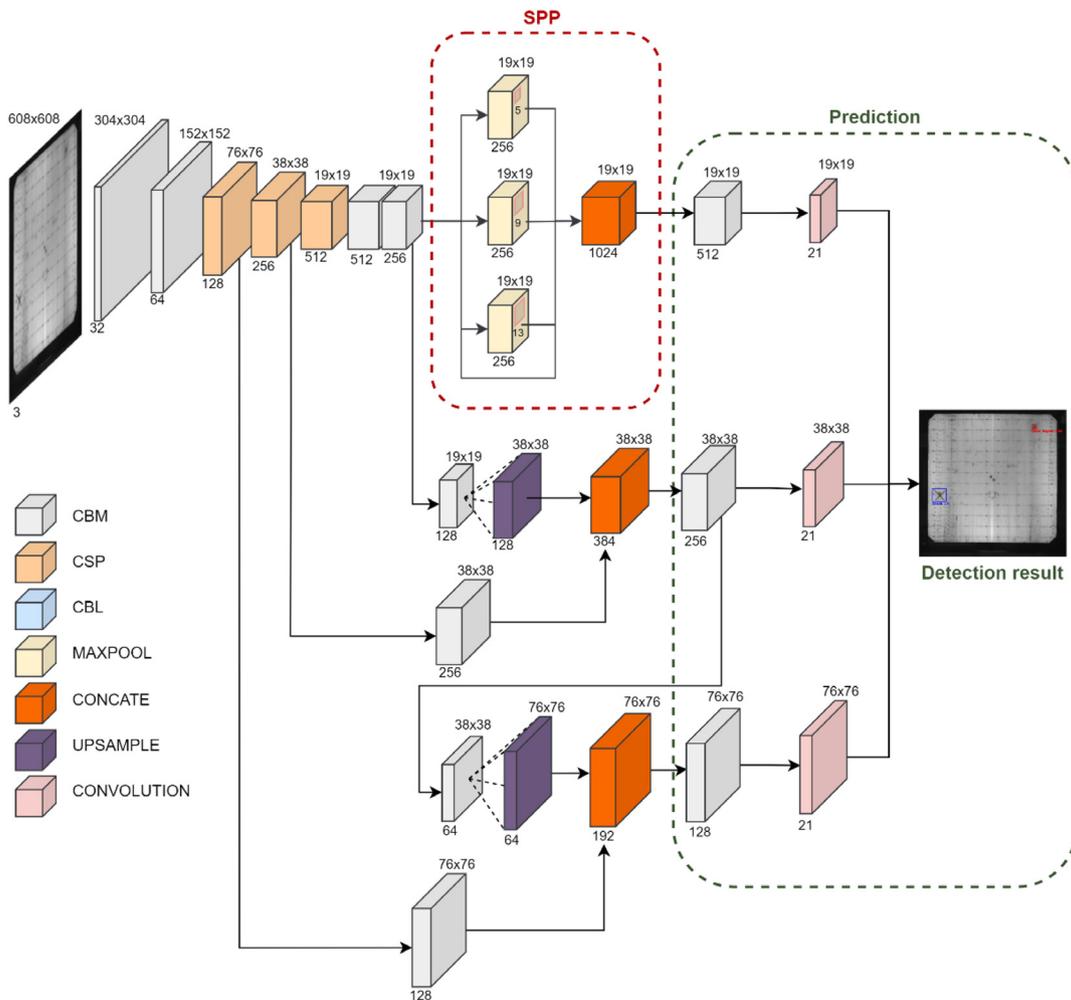


Fig. 9. Modified multi-class Tiny-YOLOv4 with SPP extractor.

Table 4. Performance of the Tiny-YOLOv4 comparing original and modified networks.

Model	Network	mAP (validation) (%)	mAP (test) (%)	mAP (average) (%)	Prediction time (ms)	Training time (h)	Memory size (MB)
Single class (microcrack)	Tiny	99.3	99.6	99.4	26.9	2.9	22.4
	Modified	99.6	98.7	99.2	27.4	3.0	35.9
Single class (dark region)	Tiny	93.7	71.4	82.6	25.8	3.0	22.4
	Modified	85.8	69.7	77.7	25.5	3.1	35.9
Multi-class (Microcrack and dark region)	Tiny	94.6	85.7	90.1	27.8	5.0	23.3
	Modified	95.7	88.1	91.9	28.2	5.3	36.8

of mAP averaging at 95.7% and 88.1% for validation and test samples respectively. The same trend is not repeated for single class models. In terms of mAP there is no significant difference between original and modified models, in particular the microcrack model. In fact the modified model for dark region has registered a slight reduction in mAP. This indicates the difficulty in detecting dark region due to complexity of such a defect. In summary both original YOLOv4 and modified Tiny-YOLOv4 are sufficiently accurate models for inspecting microcrack defect only. In contrast the original YOLOv4 is preferred for inspecting dark region defect compared to Tiny-YOLOv4 or its modified version. In case of multi-class inspection, the modified Tiny-YOLOv4 is preferred choice for an online inspection due to its speed advantage. In this case the original YOLOv4 can be used for sampling where precision and accuracy are an utmost importance.

5 Conclusion

This paper investigates solar cell defects detection using deep learning approach based on YOLOv4 framework. Various models with different configurations and parameter settings are trained for detecting microcrack and dark region defects. Overall, the original heavy-weighted YOLOv4 is the best algorithm for both single and multi-class solutions with mAP ranging from 94% to 100%. However, this algorithm is very computationally intensive since it requires at least 42 ms to inspect one sample. Also it requires high-end computers since the algorithm needs 244 MB of memory space in order to function reliably. In comparison a modified version of Tiny-YOLOv4 resulted in an accuracy of approximately 92%. Even though this algorithm has registered slightly lower mAP, however, its speed is at least twice faster compared to the original heavy-weighted YOLOv4. This speed is more competitive compared to methods published in [12,13]. In terms of accuracy the performance of the proposed model is also comparable if not better compared to [12,13]. In conclusion the modified algorithm is suitable for rapid online inspection while its heavy-weighted counterpart is more suitable for an offline application where hardware resources are abundant and speed is important but not a decisive factor. Moreover,

the algorithm can also be applied to EL inspection system since the images produced by this technology are optically similar to those generated by PL system.

Author contribution statement

Amran Binomairah is responsible in implementing most of tasks of this research project. Azizi Abdullah provides inputs on machine learning while Bee Ee Khoo contributes in image processing and artificial intelligence. Meanwhile Zeinab Mahdavi pour is responsible in preparing test samples and Teow Wee Teo involves in designing the image capturing hardware. Nor Shahirah Mohd Noor contributes in data analysis and Mohd Zaid Abdullah is the owner of the research, and the main person responsible for this paper.

The authors acknowledge TT-Vision Technologies for funding this research (304.PELECT.6050420.T148) and matching grant from Universiti Sains Malaysia (1001.PELECT.8070009).

References

1. T. Fuyuki, A. Kitiyanan, Photographic diagnosis of crystalline silicon solar cells utilizing electroluminescence, *Appl. Phys. A* **96**, 189 (2009)
2. K. Bedrich, M. Bokalic, M. Bliss, M. Topic, T.R. Betts, R. Gottschalg, Electroluminescence imaging of PV devices: advanced vignetting calibration, *IEEE J. Photovolt.* **8**, 1297 (2018)
3. W.S.M. Brooks, D.A. Lamb, S.J.C. Irvine, IR reflectance imaging for crystalline Si solar cell crack detection, *IEEE J. Photovolt.* **5**, 1271 (2015)
4. I. Zafirovska, M.K. Juhl, J.W. Weber, J. Wong, T. Trupke, Detection of finger interruptions in silicon solar cells using line scan photoluminescence imaging, *IEEE J. Photovolt.* **7**, 1496 (2017)
5. T.W. Teo, Z. Mahdavi pour, M.Z. Abdullah, Recent advancements in micro-crack inspection of crystalline silicon wafers and solar cells, *Measur. Sci. Technol.* **31**, 081001 (2020)
6. D.M. Tsai, C.C. Chang, S.M. Chao, Micro-crack inspection in heterogeneously textured solar wafers using anisotropic diffusion, *Image Vis. Comput.* **28**, 491 (2010)

7. S.A. Anwar, M.Z. Abdullah, Micro-crack detection of multicrystalline solar cells featuring an improved anisotropic diffusion filter and image segmentation technique, *EURASIP J. Image Video Process.* **2014**, 15 (2014)
8. D.-C. Tseng, Y.-S. Liu, C.-M. Chou, Automatic finger interruption detection in electroluminescence images of multicrystalline solar cells, *Math. Probl. Eng.* **2015**, 1 (2015)
9. H. Chen, H. Zhao, D. Han, K. Liu, Accurate and robust crack detection using steerable evidence filtering in electroluminescence images of solar cells, *Opt. Lasers Eng.* **118**, 22 (2019)
10. D.-M. Tsai, S.-C. Wu, W.-C. Li, Defect detection of solar cells in electroluminescence images using Fourier image reconstruction, *Solar Energy Mater. Solar Cells* **99**, 250 (2012)
11. S. Deitsch et al., Automatic classification of defective photovoltaic module cells in electroluminescence images, *Solar Energy* **185**, 455 (2018)
12. X. Li, Q. Yang, Z. Lou, W. Yan, Deep learning based module defect analysis for large-scale photovoltaic farms, *IEEE Trans. Energy Convers.* **34**, 520 (2019)
13. W. Tang, Q. Yang, K. Xiong, W. Yan, Deep learning based automatic defect identification of photovoltaic module using electroluminescence images, *Solar Energy* **201**, 453 (2020)
14. M.W. Akram et al., CNN based automatic detection of photovoltaic cell defects in electroluminescence images, *Energy* **189**, 116319 (2019)
15. L. Liu, Y. Zhu, M.R. Ur Rahman, P. Zhao, H. Chen, Surface defect detection of solar cells based on feature pyramid network and GA-faster-RCNN, in *2019 2nd China Symposium on Cognitive Computing and Hybrid Intelligence (CCHI)* (2019), pp. 292–297
16. X. Zhang, Y. Hao, H. Shangguan, P. Zhang, A. Wang, Detection of surface defects on solar cells by fusing multi-channel convolution neural networks, *Infrared Phys. Technol.* **108**, 103334 (2020)
17. U. Otamendi, I. Martinez, M. Quartulli, I.G. Olaizola, E. Viles, W. Cambarau, Segmentation of cell-level anomalies in electroluminescence images of photovoltaic modules, *Solar Energy* **220**, 914 (2021)
18. J. Balzategui, Eciolaza, D. Maestro-Watson, Anomaly detection and automatic labeling for solar cell quality inspection based on generative adversarial network, *Sensor* **21**, 1 (2021)
19. P. Kunze, S. Rein, M. Hemsemdorf, K. Ramspeck, M. Demant, Learning an empirical digital twin from measurement images for a comprehensive quality inspection of solar cells, *Solar RRL* **6**, 2100482 (2022)
20. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2016)
21. A. Bochkovskiy, C.-Y. Wang, H.-Y.M. Liao, YOLOv4: Optimal Speed and Accuracy of Object Detection (2020)
22. C.Y. Wang, A. Bochkovskiy, H.Y.M. Liao, Scaled-yolov4: scaling cross stage partial network, in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2021), doi:[10.1109/CVPR46437.2021.01283](https://doi.org/10.1109/CVPR46437.2021.01283)
23. K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* **37**, 1904 (2015)

Cite this article as: Amran Binomairah, Azizi Abdullah, Bee Ee Khoo, Zeinab Mahdavi-pour, Teow Wee Teo, Nor Shahirah Mohd Noor, Mohd Zaid Abdullah, Detection of microcracks and dark spots in monocrystalline PERC cells using photoluminescence imaging and YOLO-based CNN with spatial pyramid pooling, *EPJ Photovoltaics* **13**, 27 (2022)