Infrared Thermography Based Defect Analysis of Photovoltaic Modules Using Machine Learning

By

Ovib Hassan Mobin
16321145
Tahmid Tajwar
16321051
Fariha Reza Khan
16321021
Shara Fatema Hossain
16121095

A thesis submitted to the Department of Electrical and Electronic Engineering
in partial fulfillment of requirement for the degree of
Bachelor of Science in Electrical and Electronic Engineering

Department of Electrical and Electronic Engineering
Brac University
October 2020

©2020. Brac University
All right reserved.

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material which has been accept, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name and Signature:

Ovib Hassan Mobin
16321145

Tahmid Tajwar
16321051

Shara Fatema Hossain
16121095

Fariha Reza Khan
16321021

Abstract

Solar photovoltaic (PV) has gained much attention throughout the world for clean energy production. Faults in the PV modules cause the reduction of the amount of the electricity gain from the PV systems. Detecting faults of PV modules could help to take necessary measures. In this study, Infrared thermography (IRT) is used in order to take images of PV modules which may indicate the hotspot. Later, these images are converted into datasets for a classifier to detect the hotspot of PV modules. Besides, I-V characteristics of the PV modules are also analyzed to find out the relation between hotspot & defected area. The further part of this study has been conducted by using a machine learning tool called 'YOLO: You Only Look once'. After training & testing the learner with the datasets, the outputs are validated with the IRT images of PV modules. The major gain of this study is to apply a modified real time object detection tool to understand and detect the defect of the PV module. The algorithm is capable of detecting the hotspot using the weightage file of the training phase. Result shows that with more diversified datasets the accuracy of detecting hotspot increases.

Keywords: Infrared thermography, YOLO, photovoltaic, hotspot, machine learning

Acknowledgement

We would like to thank Mr. Md. Mosaddequr Rahman, Ph.D. Professor, Department of Electrical and Electronic Engineering (EEE), Brac University, for his support, guidance and feedbacks for completion of the thesis. We would also like to thank Mr. Mohaimenul Islam, Research Assistant at the Brac University for helping to simplify and enjoy this whole thesis experience. Moreover, we like to give thanks to Mr. Atiqul Islam for his thermal camera support.

Table of Contents

List of Figures:

List of Tables:

List of Acronyms:

PV: Photovoltaic

YOLO: You Only Look Once

IRT: Infrared Thermography

IR: Infrared

CNN: Convolutional Neural Network

I-V: Current vs. Voltage characteristics

EL: Electro-luminescence

ReLu: Rectified linear unit

‘

Chapter 1

Introduction

Solar energy is said to be harmless for the environment. Moreover, a large amount of electricity manufactured around the world comes from the non-renewable energy, which will not be able to satisfy all the needs of the ever-growing consumer. Due to the longevity of solar energy & as it is produced from the direct sun light using PV cells. Higher preference of generating power from renewable sources has led to an increase in solar photovoltaic (PV) system installations worldwide. Power generation of such systems is affected by factors that can be identified early on through efficient monitoring and detection systems. By following the modified methods, the safety of a long-term investment can be ensured. Due to the long-term matter of investment in Solar PV system, the prevention and detection of faults in a PV module is very important. There are many defects that can appear during the PV modules lifetime. Some of them have origin in the manufacturing process and others can be caused during operation and maintenance. Also, there are different tools to identify those faults on PV module. Moreover, different modern algorithm and approaches are applied to detect the faults, their location and classifications of PV modules. This study incorporates an infrared thermal image-based defect detection using machine learning based tool which is called You Only Look Once: Unified, Real-Time Object Detection "YOLO". There is lack of study for PV module fault detection using YOLO.

1.1 Motivation

Over the past decade, a rapid growth of photovoltaic (PV) technologies and systems has been observed over the world. The PV is considered as the most promising and major renewable energy source due to providing a clean energy. Solar power has become the World's most attractive and, in some cases, the cheapest form of new electricity generations [1]. PV operation can be affected by different factors such as maximum power point tracking error, environmental effects like shading and dust or snow accumulation on the PV surface, wiring losses and ageing, post manufacturing defects and malfunction in other PV components like power conditioner unit, and the inverter. A study [2] in 2010 showed that faults have a potential effect on reducing the power generated by a PV system annually by about 18.9%.

Therefore, it was essential to develop proper techniques that can detect existing and developing faults. Having a detecting algorithm can quickly detect a fault presence in the PV modules and identifies the cause behind that fault. In recent years, different detection techniques have been considered in the literatures to detect and identify various faults for PV systems. This suggests that there is no comprehensive fault detection technique capable of detecting all of the defects in a PV system [3]. It justifies the great interest that is gaining for the methods of hotspot detection and methods for diagnosing faults with non-destructive techniques, such as for example: infrared thermography.

1.2 Background

Infrared thermography (IRT) has become a popular tool in inspecting faults in electrical and medical sector. There have been studies using IR thermography to find the faults and defects in a PV module in recent years. In principle, IRT or thermal imaging refers to nonintrusive detection and measurement of radiation in the form of two- or three-dimensional pseudo color images of temperature distribution and its imaging by means of appropriate imagers (thermal camera) [4].

In paper [5], authors' have developed an algorithm which can classify thermography-based faults. For that purpose, they have taken thermal images of 8 different panels from a PV system situated at New Delhi, India.

The paper [6] tried to prove the cost efficiency & reliability of UAV (Unmanned Aerial Vehicle) & IRT (Infrared thermography) sensor on board's fast inspection on large scale PV plants. During the work process they have kept camera tilt angle, minimum irradiance level, height & drone speed along with reflection of the sun & the drone under surveillance.

In paper [7] a novel method had been introduced in order to find faults in PV power plants by using automation for detection & classification with employment of aIRT (aerial Infrared Thermography). Here a combination of DIP (Digital Image Processing) techniques & CNNs (Convolutional Neural Networks) have helped to categorize the faults into disconnected substrings, hot spots & disconnected strings.

Paper [8] showed the system of EIIT (Electromagnetic Induction Infrared Thermography), use of FFT (Fast Fourier Transform), PCA (Principle Component Analysis) & ICA (Independent Component Analysis) algorithm for sequencing thermography & most importantly how EIIT is more useful technology over other. The detection system of PV modules of this work is comprised with visualization, rapid detection, subtle fault evaluation & quantitative detection. Using pulse & lock in heating, images of defects like hot spots, micro crack broken grid & surface impurity pollution of PV cells was captured. This paper tried to promote the use of EIIt technology in PV detection industry & hopes that by using machine vision & AI, EIIt technology could help to detect the defects of PV modules in large scale.

In paper [9], at first, they have analyzed the common faults of PV modules. Later on, concentrated on the methods which help to find faults using thermography. The faults found due to the investigations are hot spots, superheat for the presence of concentrated deposits along with superheat for the presence of dust/sand distributed on the module surface. This paper work can help find failures of PV modules using thermo graphic investigations as well as can help with pre maintenance needed for the localized heating.

In paper [10] authors have used machine learning technique through thermography to develop a binary classifier for PV panels. Here the classification is done in defective & non-defective category. Here nBayes classifier had been used to categorize the panels for diagnostic perspective as well. Using 260 test samples the accuracy for this model is 98.4%. Having low computational time, this model could be used on different systems.

Fault is actually a way for which any device moves from a state of operating to a state of non- operating. For PV systems best way to detect faults are with the help of infrared thermography. For not having the direct contact with the object, IRT is said to be advantageous [5]. It is said that with the help of IRT exact faulty place of PV (Photovoltaic) modules could be detected [6]. IRT is also said to be of big help for the detection of a great number of defects found in PV modules & it can detect & prevent the failures of PV modules as well [6]. The analysis of the defects is done with the help of thermographic camera & in the image anomalies are easily identified. [9]. A new approach has been used in quite few papers which is using the computer vision or machine learning techniques with the help of IRT images. Many modern techniques take the help of machine learning to build classifier for classifying or detecting the

defects [5], [10]. A fault classification algorithm had been developed [5] here features were extracted from IR (Infrared) images to feed into the classifier & after training & testing the classifier did validate the algorithm. An approach like combining DIP techniques & CNN algorithm to classify faults found in aIRT images [11] were used but the biggest drawback here was that the method wasn't successful in detecting hotspots but faults could be found using this combinational method. But in case of using YOLO (You Only Look Once) which is basically a CNN also a machine learning tool, for the purpose of fault detection of PV modules have not been done yet. YOLO is said to be very fast in object detection & on the full image a single algorithm is applied. High accuracy could be obtained by this classifier as well. YOLO is refreshingly simple & as here detection as a regression problem is framed that is why no complex pipeline is needed but YOLO still lags behind in case of state-of-the-art detection systems in terms of accuracy [12]. A latest version of YOLO is introduced as well & it is YOLO9000. YOLO9000 is said to be a strong step towards minimizing the gap between detection & classification [12]. Because of joint training done it helped allow YOLO9000 for predicting object classes without having labeled data [12].

When using other types of machine learning approach for the purpose of finding faults on IR images it could be seen that the accuracy level wasn't satisfactory some even gave false – positives as well [11], but in the case of YOLO the more diversified the dataset is provided the detector will be more accurate on detecting.

Though YOLO is being used a lot but after going through several papers it was seen that not much work has been found using YOLO in the field of PV modules. Not much work found related to defect detection on PV module using YOLO. Moreover, this research work was needed to be done in order to apply a modern & fast detector like YOLO (You Only Look Once) in case of finding faults of PV modules by using the IR images.

1.3 Aims and Objective

The aim of this thesis is to identify the hotspot of PV modules from infrared thermal images by using machine learning algorithm. Initially I-V characteristics of the sample PV modules have been analyzed along with the infrared images to determine the hotspot region and the temperature. After that sample infrared images are characterized and prepared for training in

the machine learning tool. The main objective of this work is to use a detection tool to identify hotspot from the infrared image of PV module. For this purpose, YOLO is used in this study after necessary configuration and modification to detect the hotspot from a PV module in real time as well as from images. To train the tool 15 sample PV modules of same watt are used.

1.4 Scope of the study

The scope of this works mainly tackles with the detection of hotspot on the surface of the PV module. It also relates the I-V properties with the hotspot (i.e. significant temperature difference faults on PV modules) temperature which is done manually. Which in turn shows a relationship between performance degradation and temperature. Furthermore, YOLO, a real time object detection tool is used to automatically detect the hotspot from a thermal image. Infrared images are trained and tested after configuring and modifying the tool to perform this study. The modified tool can detect hotspot from live infrared images based on the training dataset.

1.5 Organization of the thesis

The rest of the book is organized in such an order to make it easy to understand for a reader. The book is organized as follows: Chapter 2 presents the theoretical background of photovoltaic module and Theory related to image processing, YOLO (You Only Look Once) and algorithm is discussed in Chapter 3. Chapter 4 describes the experimental set up and methodology. Chapter 5 describes the results analysis and performance evaluation of IV of PV modules for this study whereas Chapter 6 provides analysis & performance of the classifier for the hotspot detection of PV modules using machine learning platform ‘YOLO’ and discussions and finally Chapter 7 provides the Conclusion and Future Works.

Chapter 2:

Photovoltaic Module & Infrared Thermography Background

2.1 Introduction

This chapter discusses on the theoretical knowledge and information which is needed to understand the analysis of a PV module using IR thermography. As it is necessary to find a proper testing method for PV module for which is very important to have a clear concept regarding the Photovoltaics and Infrared Thermography.

2.2 Photovoltaic module

A PV module is an assembly of photovoltaic cells mounted in a framework. Sun is the main source of energy which helps the photovoltaic cells to generate direct current electricity. In a PV module, the PV cells are wired in parallel to produce current and in series to provide a higher voltage. The front part of the module is made up of tempered glass and a protective water-resistant material on the back. The edges are sealed. An aluminum frame holds everything together in a mountable unit. There is a junction box at back of the module which holds all the electrical connections. An equivalent circuit of a PV cell is showed in the fig 2.1.

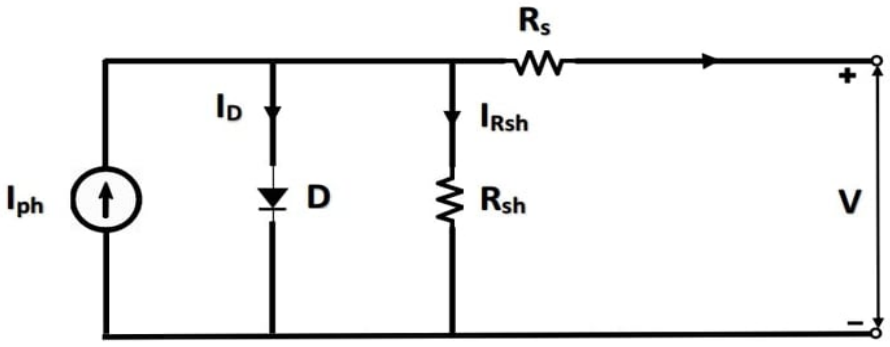


Fig 2. 1 Equivalent circuit of PV cell.[13]

Here, the constant current source I_{ph} represents the light-induced current produced in the cell is in parallel with the p-n junction diode. Rs represents series resistance and R_{sh} represents the shunt resistance. The output current (I) can be demonstrated in the following equation.

$$I = I_{ph} - I_D - I_{RS} \text{---------- (2.1)}$$

Besides, to improve the voltage the PV module is connected in parallel connection. From here "I" can be calculated for a PV panel. An equivalent circuit diagram of a PV module is displayed below.

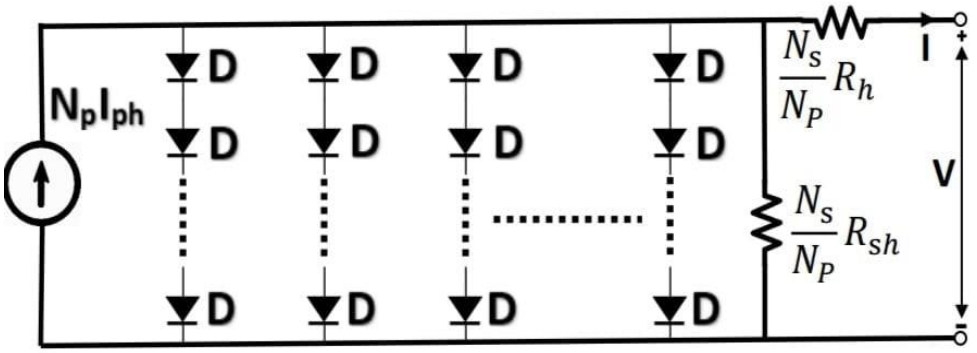


Fig 2. 2: An equivalent circuit diagram of a PV module.[13]

$$I = N_p I_{ph} - N_P I_S \left(e^{\frac{q(N_p V + N_S I R_S)}{N_p N_S A K T_C}} - 1 \right) - \frac{\frac{N_p}{N_S} V + I R_S}{R_{sh}} \text{---------- (2.2)}$$

From the above equation, $I_s$is the reverse saturation current, V is the output voltage, q is the charge of an electron, n is the ideality factor of the p-n junction, k is the Boltzmann's constant Np is the number of cells connected in parallel and Ns is the number of cell connected in series in the module.

When differentiating the types of solar panels, 3 types could be found based on performance, cost & installations. They are monocrystalline, polycrystalline & thin film. The features of monocrystalline & polycrystalline almost match with each other but in case of monocrystalline panels silicon is formed into bars & cut into wafers on top of that this type is said to be most efficient & a bit expensive compared to others. In the case of polycrystalline while forming the

wafers fragments of silicon are melted together. The reason for this solar panel to be less efficient is that this space in this panel is limited for electrons to move freely. Another type is the thin film solar panel & this panels are made by depositing one or more thin layers, this type of panel is said to be the second-generation solar cell.

In this study due to low cost polycrystalline solar modules are used.

Table 2. 1: Types of different solar panels & their merits & demerits

Type of Solar Panel	Merits	Demerits
Monocrystalline	Elevated performance	Very costly
Polycrystalline	Low cost	Inadequate Performance
Thin-Film	Flexible & transportable Lightweight	Little productivity

2.2.1 Effect of temperature

As this is known earlier that photovoltaic solar panels convert sunlight into electricity. From this concept this thought might come into head that the more sunlight the cell absorbs, the performance will be better. But this concept is not always true. Because sunlight not only consist of light that is visible but also of invisible infrared radiation which carries heat. The solar panels will perform great if it gets a lot of light. But as it gets hotter, its performance degrades. The solar cell performance can be understood from the IV curves as shown in fig 2.3 [14]. When there is no load or zero resistance, the device can produce maximum current and zero voltage which referred to short-circuit current (Isc). Alternatively, when there is many load or infinite resistance, the device produces maximum voltage and zero current, referred to open circuit voltage (Voc). In semiconductor band gap is an important term which is the energy difference between valance band and conduction band. The short circuit current (Isc) increases slightly with temperature as the band gap decreases on the other hand, the open circuit voltage (Voc) being less affected by temperature.

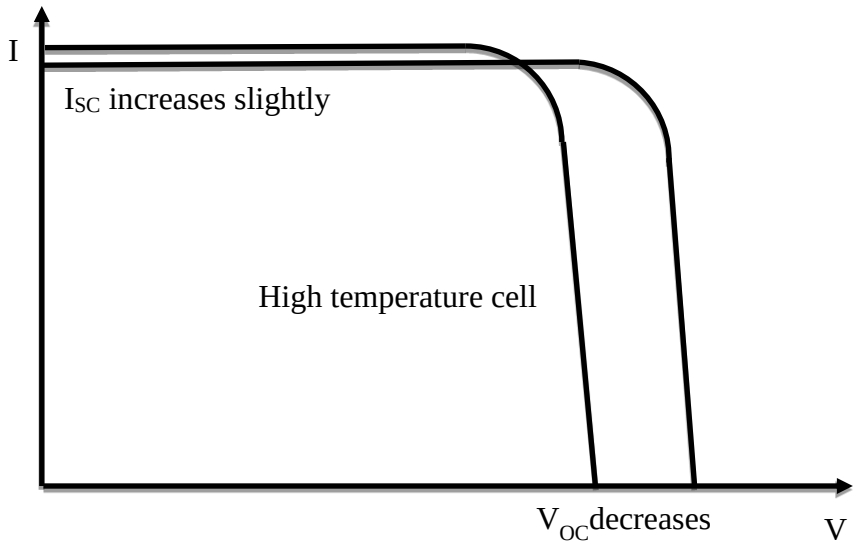


Fig 2. 3: Effect of temperature on module short circuit current and open circuit voltage.

2.2.2 Fault of PV modules

The performance of PV modules reduces due to faults & causing the loss of power a majority of these faults are undetected. As faults can lead to huge amount of power loss as well as a concern for safety issue, that is why knowing about types of faults are the first stage of fault analysis. For detecting faults there are several ways & they are stated in the table below:

Table 2. 2: Methods of fault detection [15]

Method	Process
Visual	Discoloration, surface soiling, browning,
Thermal	Thermal extraordinary heating
Electrical	Illuminated I-V curve measurement, Transmittance line diagnosis

Some faults are found which turn out to be very common in most of the PV modules, these faults are due to the structure of the PV modules, and they are shown in the table below-

Table 2. 3: Some of the most common faults in PV modules [15]

Fault type	Description
Series Arc Fault	Results from solder disjoint, cell damage, corrosion of connectors
Parallel Arc Fault	Due to breakdown of insulation of current carrying conductors
Bypass Diode Fault	Due to incorrect connection short circuit
Open circuit Fault	Object falling on PV panels, loose termination of cables, plugging & unplugging connectors at junction box
Degradation Fault	Increase of internal series resistance due to yellowing & browning, delamination, cracks in cells & interconnection

Despite the faults stated above many of the faults could be detected through visual inspection. Visual inspections are said to be easier as it could be done in the laboratory as well as before or after the testing of PV modules. Below in the table are stated some typical faults found through visual inspections.

Table 2. 4: Failures found while visual inspection [9]

Faults on the PV module component	Failures of PV modules
Wire – Connectors	Exposure of electrical parts, detachment
PV cells	Cell which are broken, cracked
Front side of PV modules	Bubbles, browning
Junction box	Loose, oxidation
Frame	Misaligned, curved, shattered

2.3 Infrared Thermography

In simple words, infrared thermography is a method which detects infrared energy emitted from object converts it into temperature and displays image of temperature distribution. Here temperature is measured without contacting an object. It is stated in Plank's law that, in thermal equilibrium at a given temperature T, the spectral density of electromagnetic radiation emitted by a black body when there is no net flow of matter on energy between the body and its environment. [17] Based on the Plank's law information, it is conveyed that hot objects emit more of their light at short wave lengths and cold objects emit more of their light at long wave lengths. This idea is used in infrared thermography. As a result, the mid wave (7 to 14 µm) infrared wave sensor is used to detect infrared energy emitted from an object. After this, infrared thermography prepared a surface temperature map of an object.

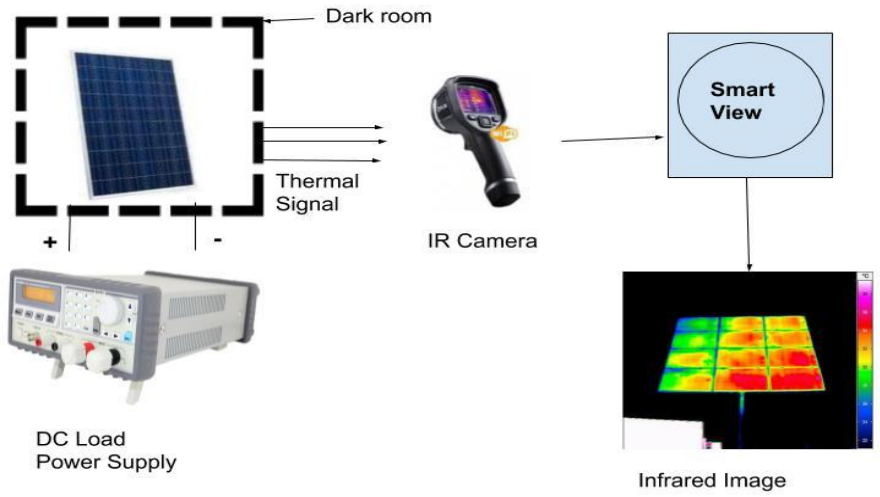


Fig 2. 4: Steps of having infrared image of PV modules.

At this moment, various process of detecting faults in PV module are available. One of the ways is Electroluminescence (EL) imaging. This technique developed to detect the radioactive recombination of charge carriers excited under forward bias. With the help of EL technique silicon solar cell properties can be categorized like minority carrier diffusion length and series resistance, broken fingers, dark dots and inactive cell but mostly for localization of crack. [18] In EL process the faulty or defected part is displayed as dark region as the resulting light intensity is proportional to the voltage of the normal EL image. [19] Alternatively, due to some limitation

like optical degradation and failure for example delamination up glass (front cover) breakdown EL technique for diagnosing fault in PV module is not very much applicable.

On the other hand, IRT imaging is a valuable tool as it has acquired much importance during the last years. Various failure like cell fracture, deficient solder joints, short circuited cell and bypassed substrings can easily be identified with the help of IRT imaging tool. As IRI imaging technique has many advantages like fast, portable, non-contact, inexpensive and are capable of large area inspection. Hence in this thesis work IRI method is used. The steps of IR thermography are showed in Fig 2.5.

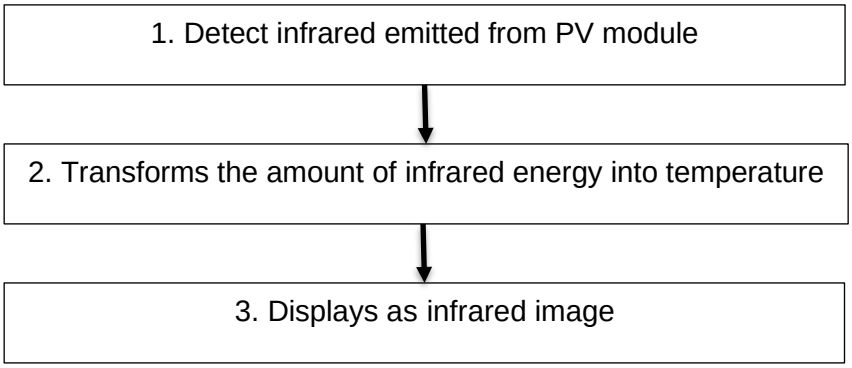


Fig 2. 5: Steps of IR thermography.

Basically, IR thermography has two different approaches. They are passive and active approach. In case of passive thermography, the background is different from the interested area which are naturally at higher or lower temperature. Here the key parameter is the temperature difference compare to a reference value. A potential defect is indicated by this abnormal temperature difference which is denoted by ΔT. However, in case of active thermography significant temperature differences is obtained by injecting current into specimen to detect any kind of abnormality. [20] Active lock in the thermography technique is used to find out the defect in this paper

2.3.1 Image processing

Image processing is a significant process to improve quality of image within a short period of time. In thermography, it is necessary to contour the hotspot area in relation to the

panel area. With the help of contour tool, it is easy to identify the shape of the object by joining all the continuous points having same intensity.

Firstly, it is necessary to convert the image to a binary image (threshold image of edge detection) with the help of OpenCV. Find contours () function. Binary image means each pixel of the image is black or white. The function in OpenCV is like finding white object from black background. There are few steps to complete the process. The steps are CV2 (OpenCV Version 3.3.1), numpy (Version 1.16.2), matplotib (Version 2.1.1). Secondly, the RGB image is converted into gray scale format. At this point, noise arises in the image as it was pixelated. It is important to remove the noise from the image because it creates random variation of brightness or color information. To remove noise Fast NI means Denoising was applied. Hence, denoising plays vital role in modern image processing systems. Thirdly, according to the variance law in probability theory, if most similar pixels are found then the noise standard deviation is low. For this the similar pixels should be arranged. This is not necessary that all similar pixels are near. It is observed in periodic patterns or elongated edges which appear in most images have similar pixels that do not stay together. Hence, the whole window around each pixel is used to find the most resembling pixel. The parameter of FastNIMeansDenoising are shown in Table 2.1.

Table 2. 5: Parameters of FastNIMeansDenoising function

Parameters	Values
1. Template window size	7
2. Search window size	21
3. Filter strength (h:)	10
4. Color component	10

Pixels in Template window size patch to compute weights. In Search window, pixels that is used to compute weighted average for given pixel should be odd. Greater search windows size means greater denoising time. h: 10, filter strength for luminance component. Bigger H removes more noise but also removes image details.

Thresholding is an easy method which is used for segmenting images. Here, each pixel is converted into either black or white pixel. The pixel will be converted into a black pixel if the

image intensity is less than some fixed constant value, or a white pixel if the image intensity is greater than that constant. To complete this process the function CV2 threshold is used. Firstly, the source image has to be converted into a grey scale image. Secondly, the pixel values will be classified using the threshold value. Thirdly, the given constant value MaxVal is to be given which will classify the pixel value is more or less than the threshold value. Finally, to separate the pixels binary inverted threshold process is used.

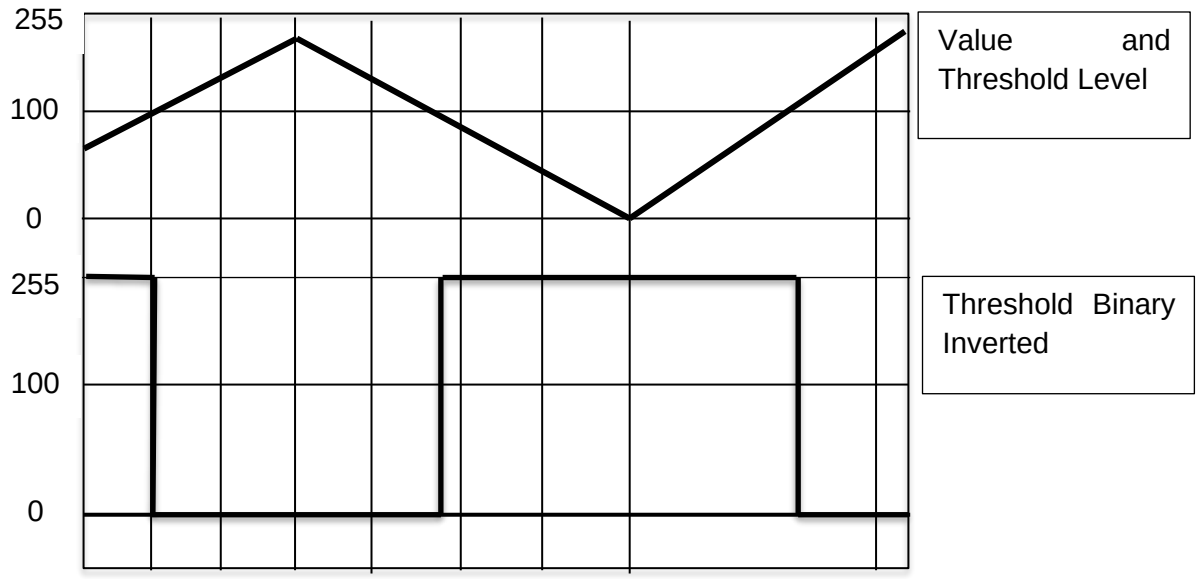


Fig 2. 6: Binary inverted threshold to separate the pixels.

The above graph can easily be explained with the help of this function:

$$dst= (x, y) = \{0 \text{ if src } (x,y) > 100 \text{ maxval otherwise}\}$$

Since the conversion of gray scale image was done from source image, so cv2.threshold function helped to be done with the process. The parameters that were used are given below:

Few parameters were used while working with cv2.threshold function. For the purpose of thresholding value 100 was taken since it could give the higher red pixel values as part of this work. MaxVal was kept as 255. In case of separating the pixels binary threshold inverted process was utilized. Finally, another function was used which was the cv2.findContours. It helped in detecting the threshold contours. To get the output of this function few specifications were

chosen. An image, detected contours as well as hierarchy of the contours are found from the function as a return.

And then, contour present in the image is identified by using cv2. find contour function. This function no longer modifies the source image but returns a modified image detected contours and hierarchy of the contours. The parameter used in the function are:

- Using threshold image as a part of input or source image
- Outer contours are restored using the RETR-EXTERNAL
- An algorithm known as CHAIN-APPROX-SIMPLE is utilized here which is responsible for providing only the end points & all the vertical, horizontal & diagonal segments are condensed. For example: a triangle contour can be encoded using three points.

These processes helped in with the calculation of the total area.

2.4 Summary

First of all, the working principle of solar module is explained with its equivalent circuit diagram and ideal IV characteristics. After that, three different types of solar modules are discussed to understand the different properties of each module. Then, different faults of solar modules are widely discussed in this section. In the last part, theory of image processing to find contour of hotspots and overall working steps are shown in this chapter.

Chapter 3

YOLO Theoretical Background

3.1 Introduction

In this section of the paper, working principle of YOLO will be discussed. Starting with the basic constituents of a convolutional neural network (CNN) and how a CNN works, laying foundation for us to understand the basic working principle of YOLO, a State-of-the-art Object detection Algorithm.

3.2 Tensors

Tensors are multi-dimensional arrays with a uniform type, they are not capable of or susceptible to change, like Python numbers and strings: contents of tensor can never be updated, only create a new one. Any scalar value is a zero-order tensor; a vector is a first-order tensor; and a matrix is a second-order tensor. A color image is made up of tensors in the order of three or it can be said a color image is third-order tensor. An image with R rows and C columns is a tensor with size $R \times C \times 3$,if a color image is stored in the RGB format, it has 3 channels for R, G and B, respectively, and each channel is a $R \times C$ matrix (second-order tensor) that contains the R or G, or B values of all pixels. A visual representation of three axis tensor shape is shown in Fig 3.1 [21]

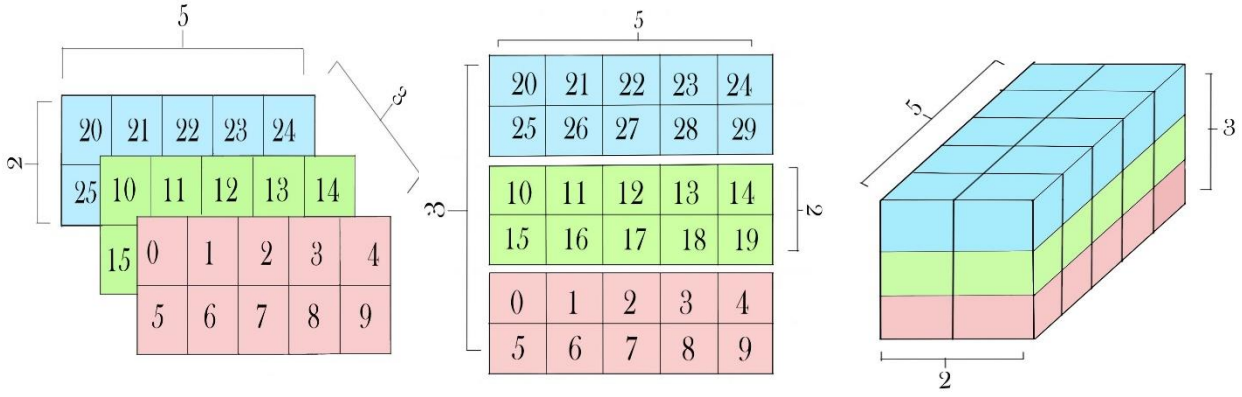


Fig 3. 1: Three axis Tensor Shape.

It is beneficial to represent images or other types of raw data as a tensor. In earlier computer vision and pattern recognition systems, a color image which is a third-order tensor, was often converted to the grayscale version, which is a matrix. But during this process the color information is lost during conversion. Color is a very important property in various image or video-based learning and recognition problems, and it is better to process color information in a principled way e.g., using a CNN (convolutional neural network). Tensors are essential in CNN. Starting with the input, intermediate representation, and also parameters in a CNN, are all tensors. Tensors with order higher than three are also widely used in CNNs. For example, the convolution kernels in a convolution layer of a CNN form a fourth-order tensor.

3.3 CNN (Convolutional Neural Network)

A CNN normally takes an order 3 tensor as its input e.g., an image with *R* rows, *C* columns, and 3 channels (R, G, B color channels) [22]. However, a CNN can handle a higher order tensor in a similar way. The input then serially goes through a number of processes. One of the processing steps is normally called a layer, which could be a convolution layer, a normalization layer, a pooling layer, a fully connected layer, a loss layer, etc.

3.3.1 LOSS Layer

The last layer is a loss layer. Let us suppose $\boldsymbol{t}$ is the target value for the input $\boldsymbol{x}^1$, then a loss function can be used to measure the difference between the prediction $\boldsymbol{x}L$ and the target $\boldsymbol{t}$. For example, a simple loss function can be

$$z = \frac{1}{2} \| t - x^L \|^2 \quad \text{..... 3.1}$$

although more sophisticated loss functions are usually used. This squared $l2$ loss can be used in a regression problem. In a classification problem, the cross-entropy loss is often used.

3.3.2 ReLU Layer

ReLU: Rectified Linear Unit for a non-linear operation. The work of ReLU is to increase the nonlinearity of the CNN to which it is used. Since the semantic information in an image (e.g.,

a dog and cat sitting beside one another Infront of a truck) is clearly a highly nonlinear mapping of pixel values in the input, we want the mapping from CNN input to its output to be also highly nonlinear. The ReLU function, is ultimately a nonlinear function.

The ReLU layer will set all negative values to 0, which means that *the probability mass function* will be *activated* only for images possessing these patterns in that particular region. This property is useful for recognizing and understanding complex patterns and objects. For example, it is a very weak evidence to support the input image contains a cat" if a feature is activated and that feature's pattern looks like a cat's head. However, if we find many activated features after the ReLU layer whose target patterns correspond to cat's head, torso, fur, legs, etc., we have higher confidence (at layer $l + 1$) to say that a cat probably exists in the input image [22].

3.3.3 Convolution Layer

Convolution is the 1st layer to extract features without losing data from an input image. Convolution preserves the correlation between pixels by learning image features using small squares of input data of the image [22]. It is basically a mathematical operation that takes 2 inputs such as image matrix and a kernel or filter. The combination of convolution kernels along with deep hierarchical structures is very effective in learning good representations such as features from images for visual recognition tasks. Convolution of an image with different kinds of filters can perform operations such as blur, edge detection and sharpen by applying filters.

3.3.4 Pooling Layer

The Pooling layers reduce the number of parameters when the images are too large. Spatial pooling which are also called subsampling or down sampling, reduces the dimensionality of each map but retains important information [22]. 2 types of pooling operators are greatly used: max pooling and average pooling. The pooling operator maps a subregion to its highest value, in case of max pooling, while the average pooling maps a subregion to its mean value.

3.3.5 Fully Connected Layer

Fully connected layer is also known as FC layer, the matrix is flattened into vector and feed into a fully connected layer like a neural network. A diagram of fully connected layer is shown in Fi 3.2 [23]

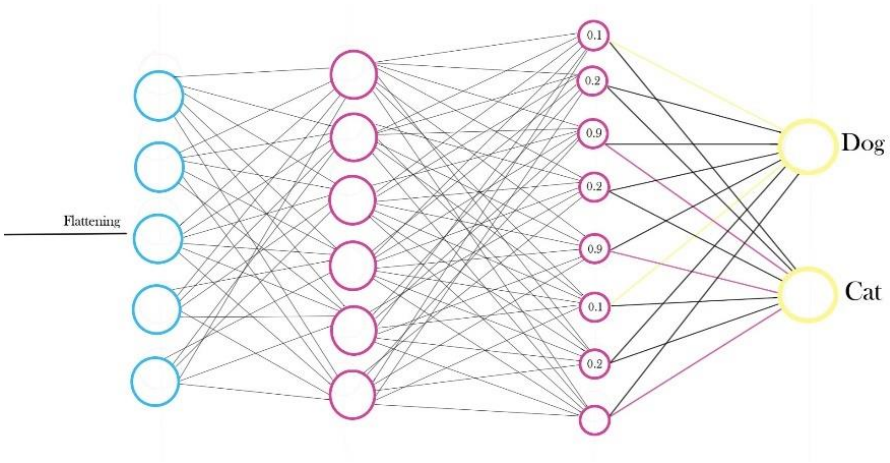


Fig 3. 2: Diagram of fully connected layer.

The feature map matrix will be converted as vector (x1, x2, x3, …). With the fully connected layers, these features are combined together to create a model. Finally, we have an activation function such as soft-max or sigmoid to classify the outputs as cat, dog, car, truck etc. Every convolutional layer is shown sequentially in Fig 3.3 [22]

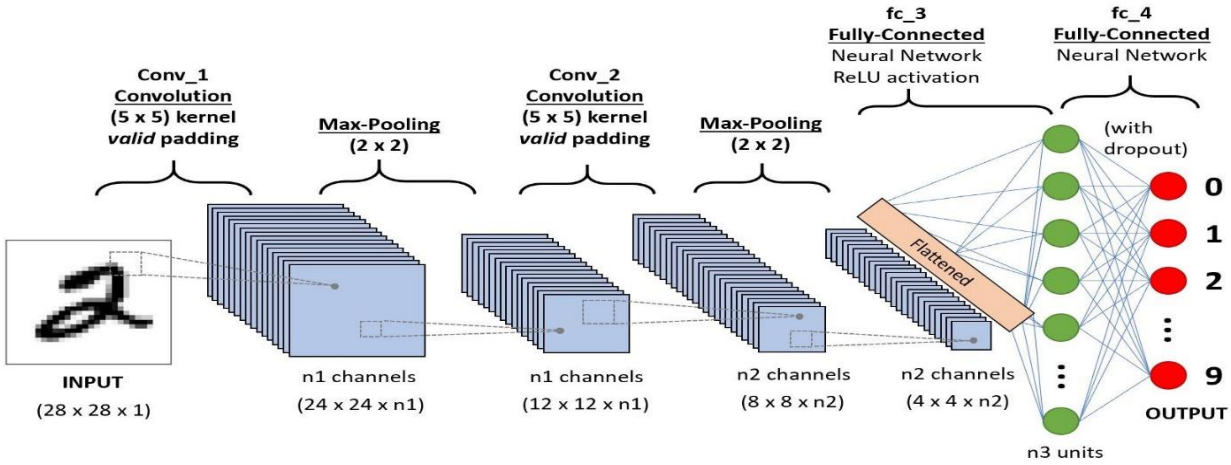


Fig 3. 3: All convolutional layer in a sequential step.[24]

3.4 YOLO (You Only Look Once)

You Only Look Once (YOLO) was developed to create a one step process involving detection and classification. Bounding box and class predictions are made after one evaluation of

the input image. YOLO predicts bounding boxes using dimension clusters as anchor boxes s shown in Fig 3.14 [25]. The network predicts 4 coordinates for each bounding box, *tx*,*ty*, *tw*, *th*.

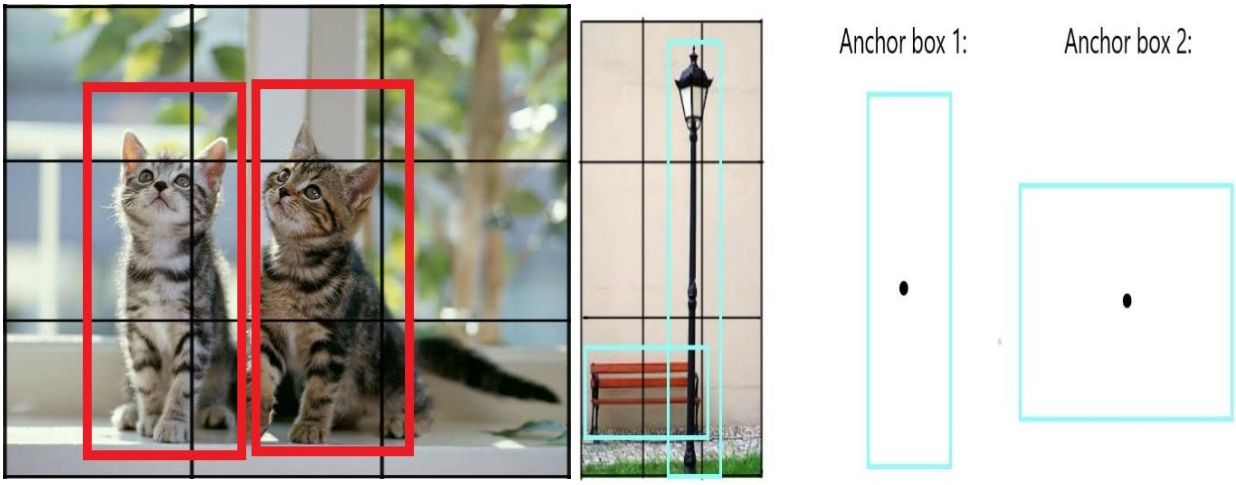


Fig 3. 4: Bounding box and Anchor box Representation.

Each bounding box can be described using four descriptors:

1. Center of the box (**bx, by**)
2. Width (**bw**)
3. Height (**bh**)
4. Value **c** corresponding to the class of an object

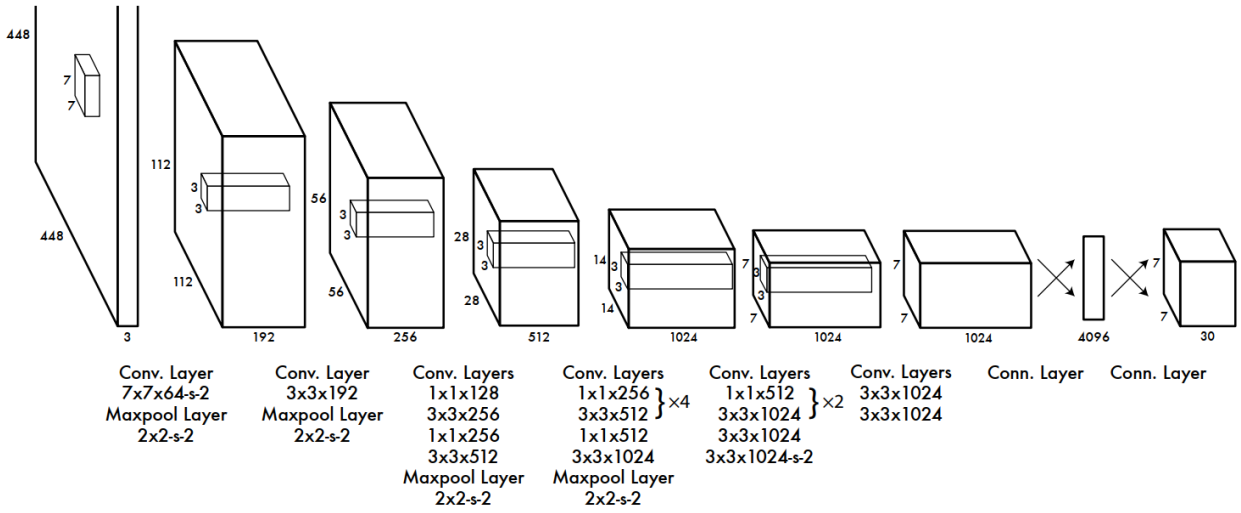


Fig 3. 5: YOLO architecture.[12]

The idea of YOLO differs from other traditional systems in that bounding box predictions and class predictions are done simultaneously. The input image is first divided into a $S \times S$ grid. Next, B bounding boxes are defined in every grid cell, each with a confidence score. Confidence here refers to the probability an object exists in each bounding box and is defined as:

$$C = Pr(Object) * IOU\ pred\ truth \quad \text{.....3.2}$$

where IOU, intersection over union, represents a fraction between 0 and 1. Intersection is the overlapping area between the predicted bounding box and ground truth, and union is the total area between both predicted and ground truth.

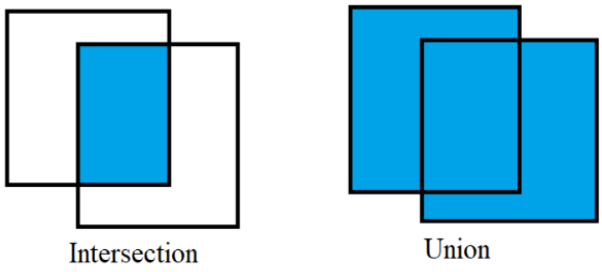


Fig 3. 6: Representation of IOU

YOLO predicts an objectness score for each bounding box using logistic regression. This should be 1 if the bounding box prior overlaps a ground truth object by more than any other bounding

box prior. If the bounding box prior is not the best but does overlap a ground truth object by more than some threshold the prediction is ignored.

YOLO uses the following equation below to calculate loss and ultimately optimize confidence:

$$Loss = \lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{A} \mathbb{1}_{ij}^{obj} [(b_{x_i} - b_{\hat{x}_i})^2 + (b_{y_i} - b_{\hat{y}_i})^2] + \lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{A} \mathbb{1}_{ij}^{obj} [(\sqrt{b_{w_i}} - \sqrt{b_{\hat{w}_i}})^2 + (\sqrt{b_{h_i}} - \sqrt{b_{\hat{h}_i}})^2]$$
$$+ \sum_{i=0}^{s^2} \sum_{j=0}^{A} \mathbb{1}_{ij}^{obj} (C_i - \hat{C}_i)^2 + \lambda_{noobj} \sum_{i=0}^{s^2} \sum_{j=0}^{A} \mathbb{1}_{ij}^{noobj} (C_i - \hat{C}_i)^2 + \sum_{i=0}^{s^2} \mathbb{1}_{i}^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2$$
..... 3.3

The loss function is used to correct the center and the bounding box of each prediction that the algorithm makes. As we have seen each image is divided into an $S \times S$ grid, with A bounding boxes for each grid. The *bx* and *by* variables refer to the center of each prediction, while *bw* and *bh* refer to the bounding box dimensions. The $\lambda coord$ and $\lambda noobj$ variables are used to increase emphasis on boxes with objects, and lower the emphasis on boxes with no objects. *C* refers to the confidence, and *p(c)* refers to the classification prediction. The 1*obj ij* is 1 if the *jth* bounding box in the *ith* cell is responsible for the prediction of the object, and 0 otherwise. 1*obj i* is 1 if the object is in cell *i* and 0 otherwise. The loss indicates the performance of the model, with a lower loss indicating higher performance. Fig 3.7, shows how it localize an object inside a box after classifying the object.

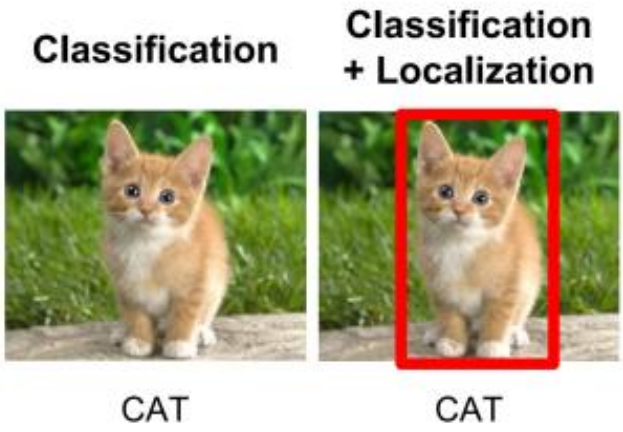


Fig 3. 7: Both classification and localization can be done through YOLO.[26]

3.5 Summary

In the end, it is certain that YOLO is a clever convolutional neural network (CNN) for doing object detection in real-time. The neural network architecture of YOLO contains 24 convolutional layers and 2 fully connected layers. The algorithm applies a single neural network to the full image, and then divides the image into regions and predict bounding boxes and probabilities for each region. YOLO is later improved with different versions such as YOLOv2 or YOLOv3 in order to minimize localization errors.

Chapter 4

Experimental Setup and Methodology

4.1 Introduction

This chapter discusses about the experimental set up and procedure of the study. Proper experimental set up is necessary to collect the reliable data. For this purpose, different hardware and software is necessary to set up. There are two hardware set up and two software is used. To collect the I-V data of the sample PV modules, one illumination box is prepared to collect the data. On the other hand, infrared thermal camera is used to obtain thermal images of sample PV modules. For this reason, one dark box is made to take the IR images of PV modules. After that to process the images, basic image processing tool is used which is SmartView software. Finally, to analyze those images with machine learning python-based environment and software is used.

4.2 I-V Measurement Setup

To perform the I-V measurement of PV modules, one wooden box is prepared with light bulbs inside the box. Fig 4.1 shows the schematic diagram of the experimental setup. The inner surface of the box is covered with reflective paper to get maximum illumination. Total ten incandescent bulbs each of 200 watts are installed as light source. Since incandescent bulb generate heat, an exhaust fan is used to keep the temperature around 25°C. Moreover, to monitor the temperature on the surface of PV module a temperature sensor is placed inside the box.

DHT11 sensor is used which can measure temperature and humidity. This sensor has an accuracy of ±1°C and it provides data in every two seconds. Fig 4.2 depicts the connection diagram of this sensor. To process the data Raspberry pi module is used and the data is showed in to a display.

Fig 4.3 shows the schematic connection diagram of the experimental setup under illumination condition. A rheostat of 100-ohm was used as variable resistive load. To measure the open-circuit voltage, switch S1 was kept open and to get the short-circuit current, both S1 and S2 switches were kept closed. In this whole experimentation temperature is kept around 25°C

and the illumination level is also kept constant by switching on the six bulbs. It is assumed that six bulbs are giving the similar irradiation of 500 W/m^2.

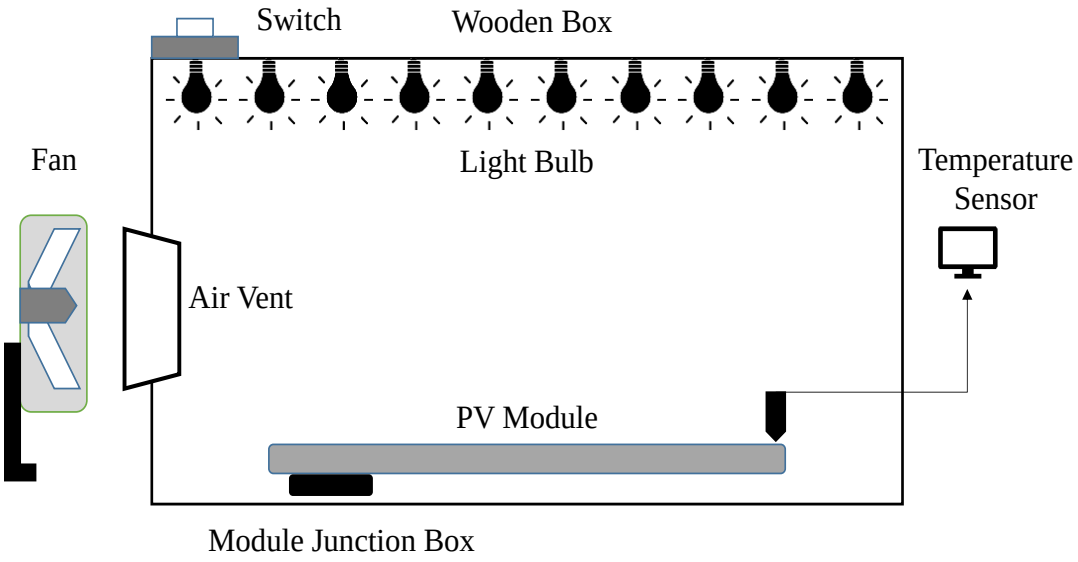


Fig 4. 1: Schematic diagram of experimental set up for I-V measurement of PV modules with constant illumination and temperature using bulbs, air vent and fan.

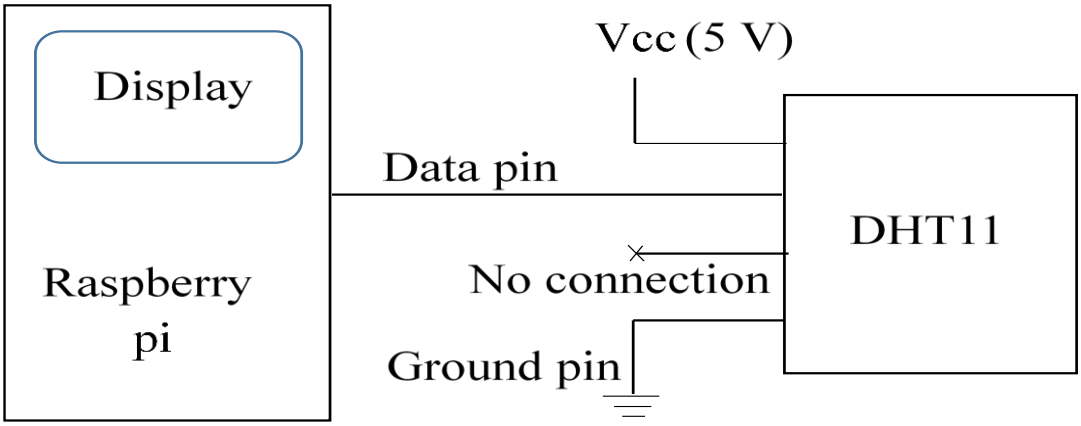


Fig 4. 2: Connection diagram of DHT11 sensor with Raspberry pi

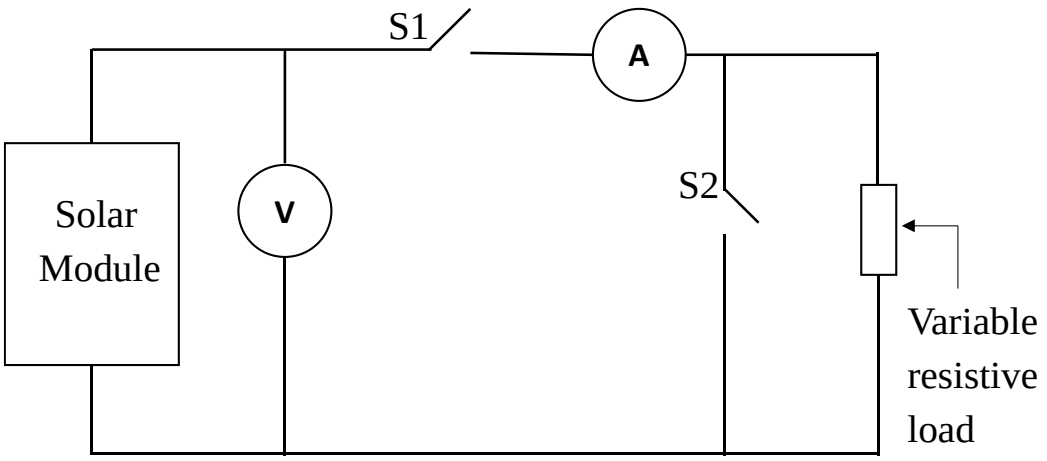


Fig 4. 3: Schematic circuit diagram for measuring IV data of each PV module. S1 is kept open for open circuit and S2 is kept open for short circuit.

4.3 Setup for Infrared Thermal Imaging of PV modules

An experimental platform was set up as shown in Fig. 4.4 to take the thermal image properly. The box is made with card board, shown as the shadowed region in the Fig, which is used to provide the dark environment for the PV module. An external DC power supply (30V/3A) is used to provide necessary voltage and current to the PV modules. A temperature sensor is used to measure the ambient temperature. In this study Fluke TiS 10 infrared camera was used to record the IR images of the modules. Table I shows the detail specifications of the thermal imaging camera.

Table 4. 1: Specifications of infrared thermal camera

Parameter	Value
IR Resolution	80*60
Thermal Sensitivity	≤150mK
Minimum Focus Distance	15cm
Spatial Resolution	7.8mRad
Image Frequency	9Hz
Accuracy	±2°C

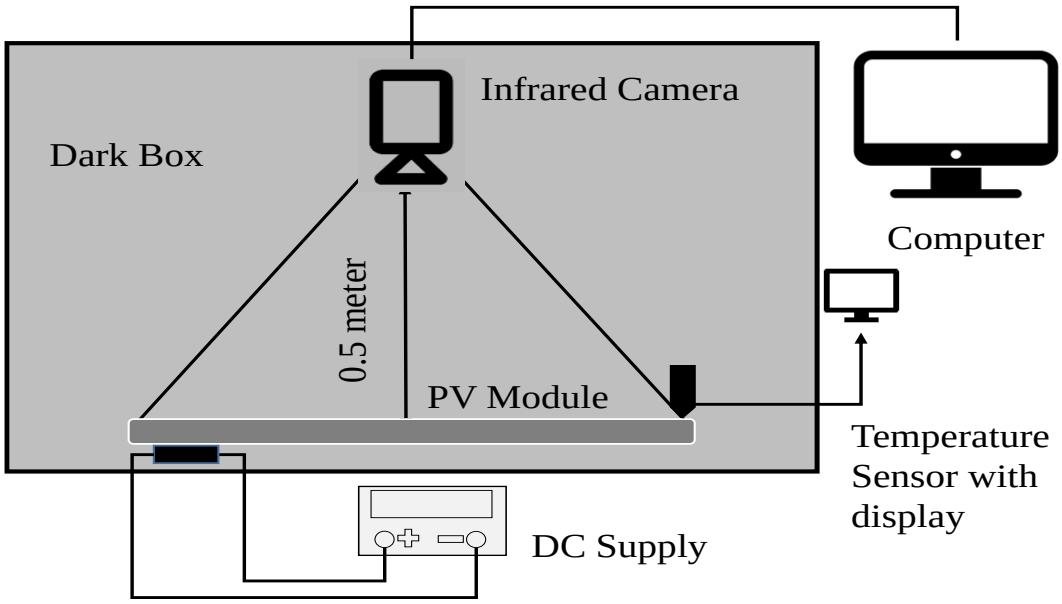


Fig 4. 4: Schematics diagram of experimental set up for taking thermal image of PV modules using infrared camera in a dark box.

Distance between the camera and the PV module was set at 0.5 meter considering the module size. Several images have been taken for each PV sample. At first module was placed inside the box. Then it was connected with the DC supply to provide necessary voltage and current for one minute. For safety purpose current value is kept lower than that of the rated short circuit current value of the sample modules. After that several images were taken for better quality. Images were taken in 1st minute, 5th minute, 10th minute, 15th minute, and 20th minute with the thermal imaging camera. All the images are processed with the computer software for further procedures.

4.4 Software Setup

In this study, a Computer with Intel “core i7” and NVDIA “RTX2070 Super” graphics card is used. And the operating system is “Windows 10” 64bit version. Since, YOLO needs high computation power for training high configuration PC is mandatory.

4.4.1 Installation of OpenCV from its Source File

First of all, OpenCV is need to install from source. OpenCV can be install directly for windows from their official website but the windows version does not work with graphic card

rather it works with CPU. Any machine learning tool works much faster with high graphics card. So, OpenCV needs to work with the graphic's CUDA cores and that is why OpenCV is installed from its source file. The source file is also available in its official website. After downloading the source file, some configuration must be done so that it can work with graphics card. To make this changes a software name "CMake" is installed and with the help of this software necessary changes can be made and generate a solution file for OpenCV.

Before making changes in CMake, CUDA must be install in the PC. Every graphics card has its own mode of "CUDA ToolKit" version. Since in this setup RTX 2070 Super is used, CUDA version for this particular model is "10.2.89_441.22". after finding the appropriate version then it is downloaded from NVDIA website as shown in Fig 4.5 it is install in to the PC.

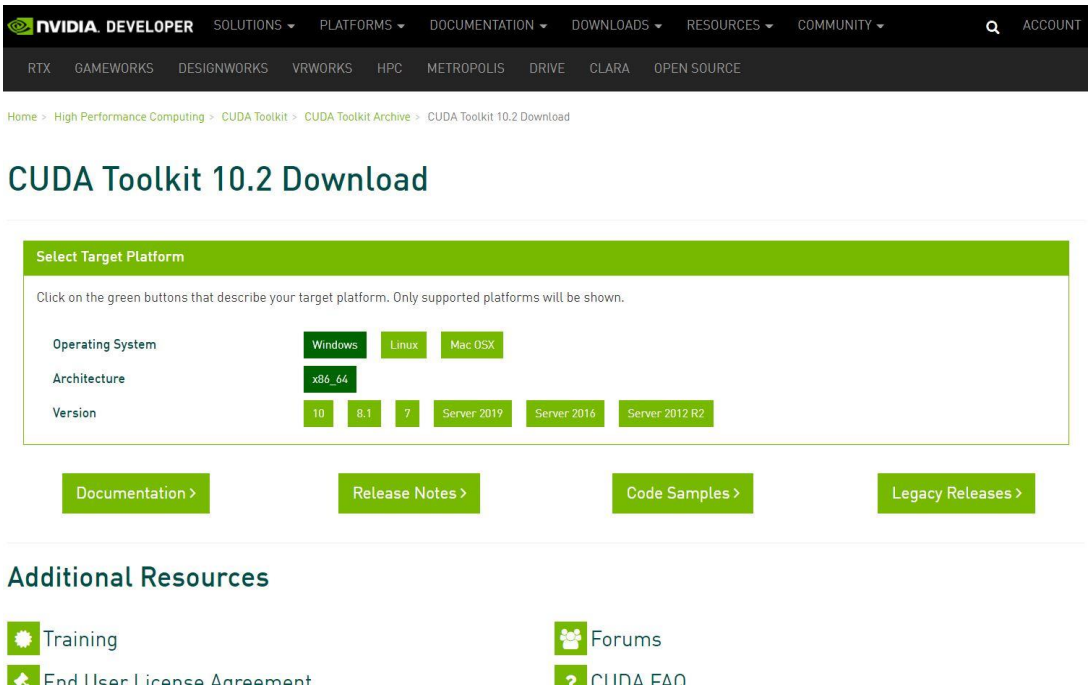


Fig 4. 5: Download CUDA from the official website of NVDIA

After successfully install CUDA into the PC, it is time to configure OpenCV in CMake. By opening CMake software, two tabs will appear where in one-tab destination of the source file of

OpenCV is given and in another tab destination of the build folder is given where CMake will generate all the OpenCV file after necessary configuration. From CMake window as shown in Fig 4.6 it seen that “WITH_CUDA” box is selected that means now OpenCV will work with all the Cuda cores of the graphics card. After configuring, generate button is clicked so that a OpenCV.sln file creates in build folder. This “OpenCV.sln” file is codified file of OpenCV, now this solution file needs to be built to operate. For building OpenCV another software is needed that is “Microsoft Visual Studio 2017”. This software is installed from Microsoft official website and after downloading, OpenCV.sln file is opened as shown in Fig 4.7. After opening the solution file “ALL BUILD” is selected which will build the actual OpenCV with CUDA compatible for the PC.

Fig 4. 6: CMake window where all the changes are made

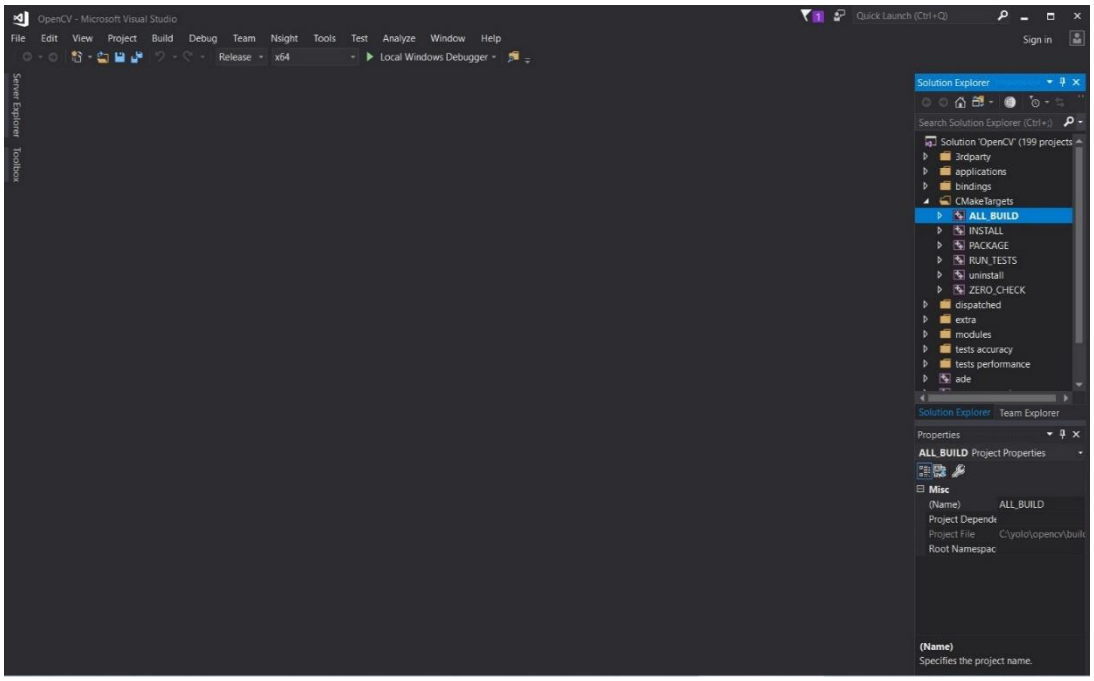


Fig 4. 7: Visual studio window after opening opencv.sln

A block diagram representation of the OpenCV setup is shown in the following Fig 4.8

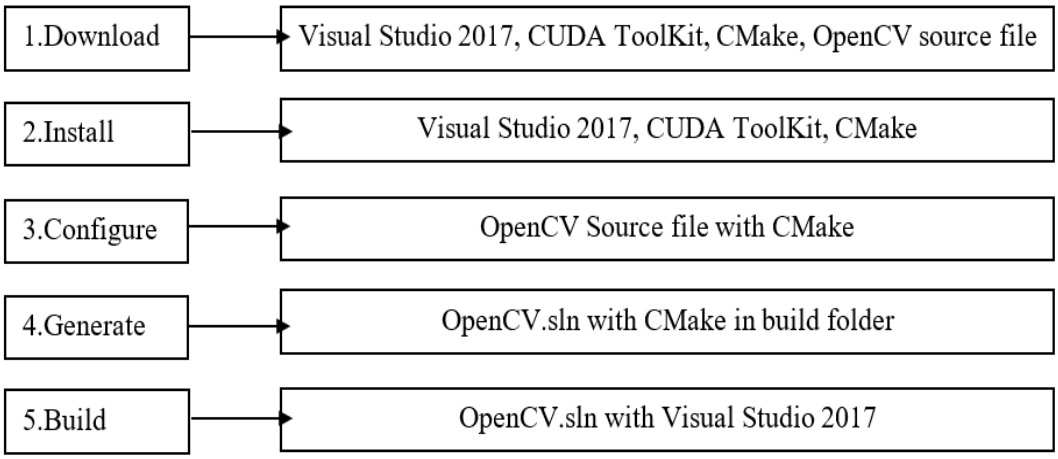


Fig 4. 8: Block diagram representation of OpenCV setup

4.4.2 Installation of YOLO

YOLO is a high-end machine learning algorithm and to run this algorithm an executable file is needed to be install. The executable file of YOLO is darknet.exe to build this application first it's source file is downloaded from GitHub. After downloading, there is a "Makefile" name file which need to be modified. Initially GPU, CUDNN value is "0" which means darknet will not work with GPU. So, it is changed to "1" in both the cases which will instruct darknet to work with GPU when it is built as shown in Fig 4.9.

After making the change, CMake software is again opened and configure darknet. Here, for the first tab the root directory of the darknet is selected and for the second tab, build folder is selected which is inside the darknet folder as shown in Fig4.10. After generating the "darknet.sln" file in build folder, again visual studio 2017 is opened and build "daknet.exe" from the solution file. Finally, darknet is now a complete usable application but yet it needs to be modified according to the study. A block diagram representation of darknet setup is shown in Fig 4.11.

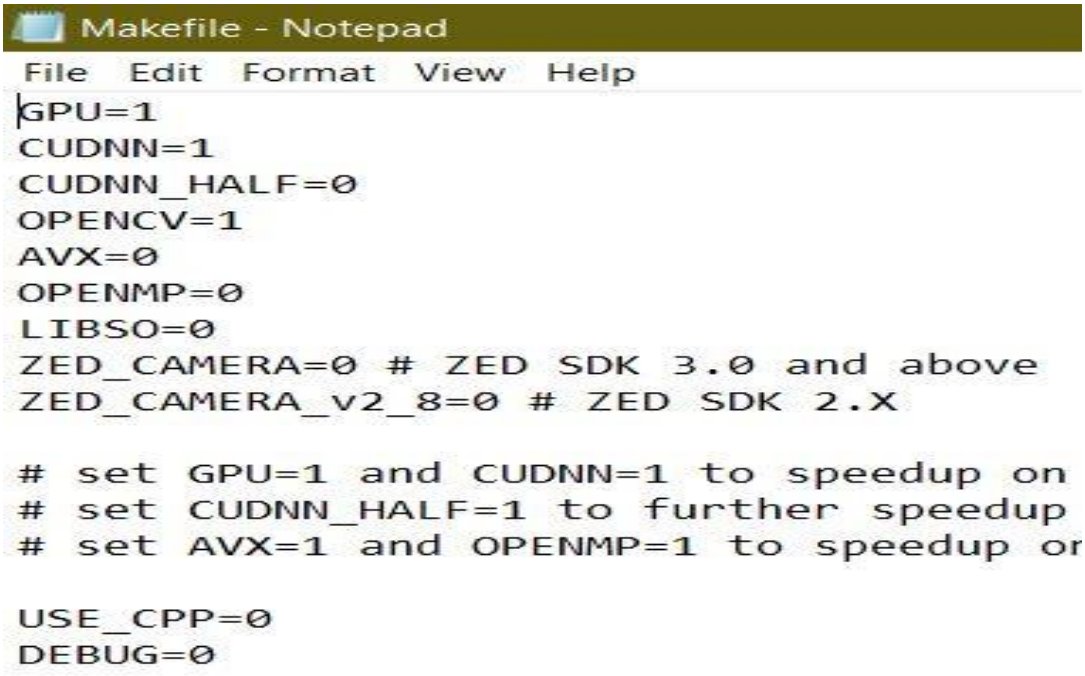


Fig 4. 9: Modification in "Makefile" of darknet

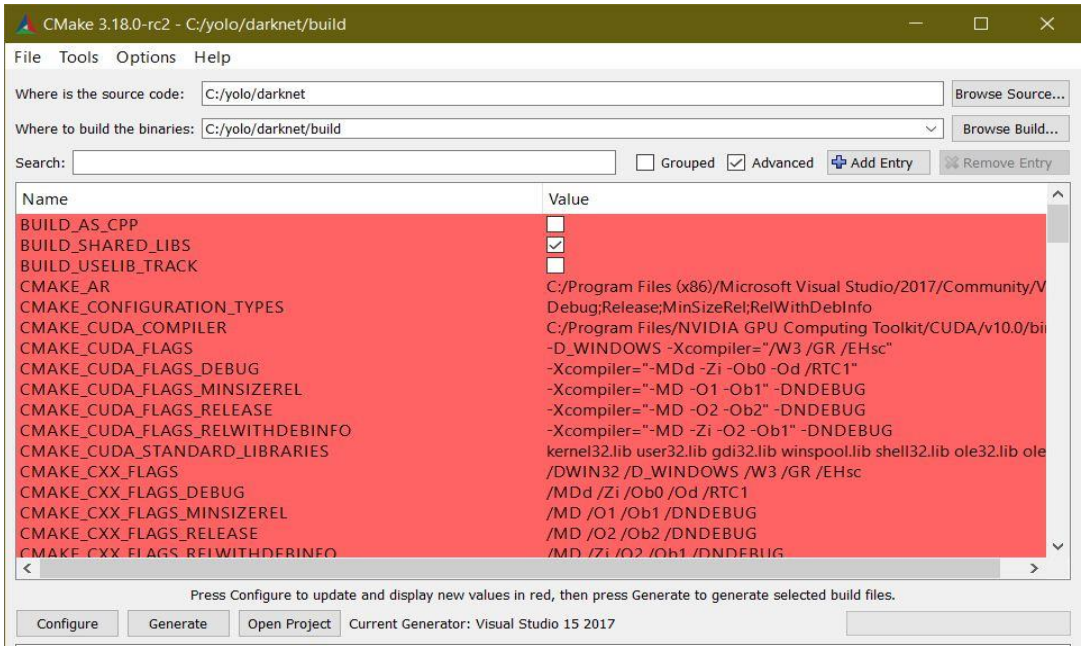


Fig 4. 10: CMake window for configuring and generating darknet

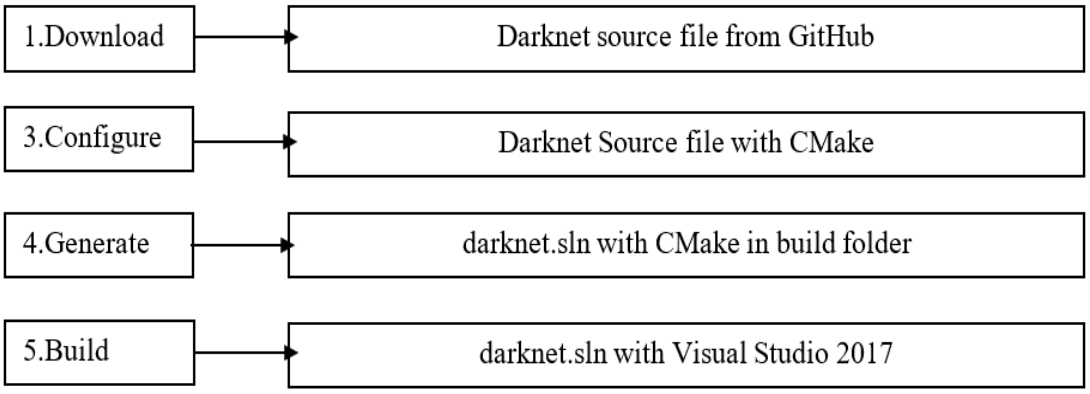


Fig 4. 11: Block diagram representation of darknet setup

4.4.3 Training of YOLO

After successfully install darknet into the pc, it's time to create custom configuration file for the detector. There is a default configuration file in the "cfg" folder inside darknet but it needs to be modified so that the training is optimized as shown in Fig 4.12

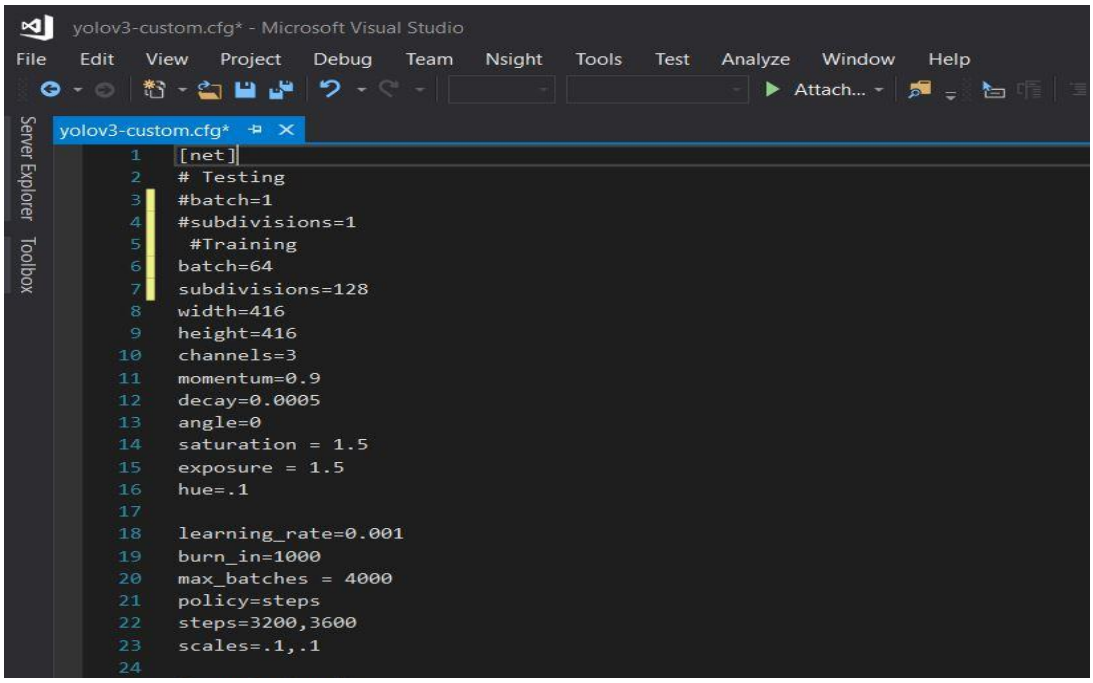


Fig 4. 12: Configuration file of YOLO

Inside the configuration file there are two mode test mode and train mode. Since the first task is to train, the test part is commented and other is uncommented. After that, batch number is set to 64 which means, 64 images are used in one iteration to update the parameters of the neural network. And subdivision is set to 128 which means, splitting the batch into mini-batches and sent to GPU for process. Next, change max batches to 4000. It is calculated by (classes*2000 but not less than 4000) and in this case there is only one class which is hotspot. So, the detector will go through 4000 times for each training image. further, step is changed to 3200, 3600 and this is calculated by (80% of the max batches and 90% of the max batches). Then, there is three [YOLO] layer in the source code and all three need to be modified. In [YOLO] layer, classes

must be set to “1” since there is only one class in this case. Furthermore, there is a [convolutional] layer before every [YOLO] layer and each [convolutional] layer must be modified. Inside [convolutional] layer value of filters must be set to “18”. This is calculated by {filters =(classes+5) *3} as shown in Fig 4.13

```
599  [convolutional]
600  size=1
601  stride=1
602  pad=1
603  filters=18
604  activation=linear
605
606
607  [yolo]
608  mask = 6,7,8
609  anchors = 10,13,  16,30,  33,23,  30,61,  62,45,  59,119,  116,90,  156,198,  373,326
610  classes=1
611  num=9
612  jitter=.3
613  ignore_thresh = .7
614  truth_thresh = 1
615  random=0
```

Fig 4. 13: Changes made inside the configuration file of YOLO

After configuring the custom yolo file, a new file is created that is “obj.names” inside “Data” folder. in this file, class name must be mentioned and in this case the class name is “Hotspot”. Next, create another file name “obj.data” inside the same “Data” folder. This file will direct the detector where the training data are located as shown in Fig 4.14. Now, inside the “Data” folder a text file name “train.txt” is created where location of each file is saved as shown in Fig 4.15.

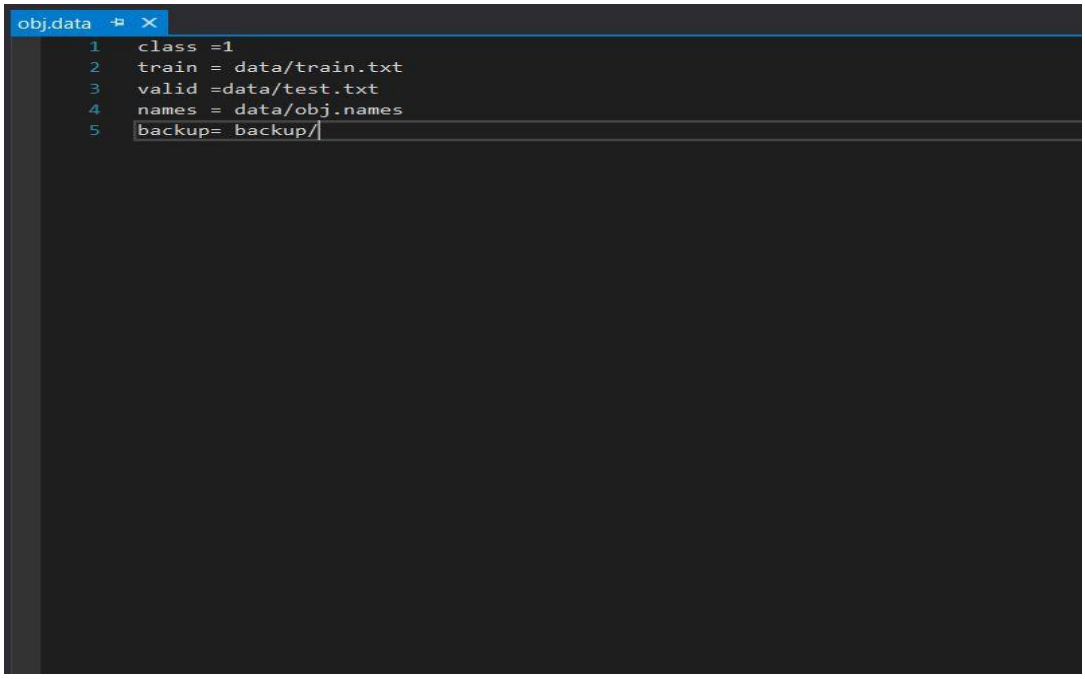


Fig 4. 14: Creating “obj.data” file to locate the training folder

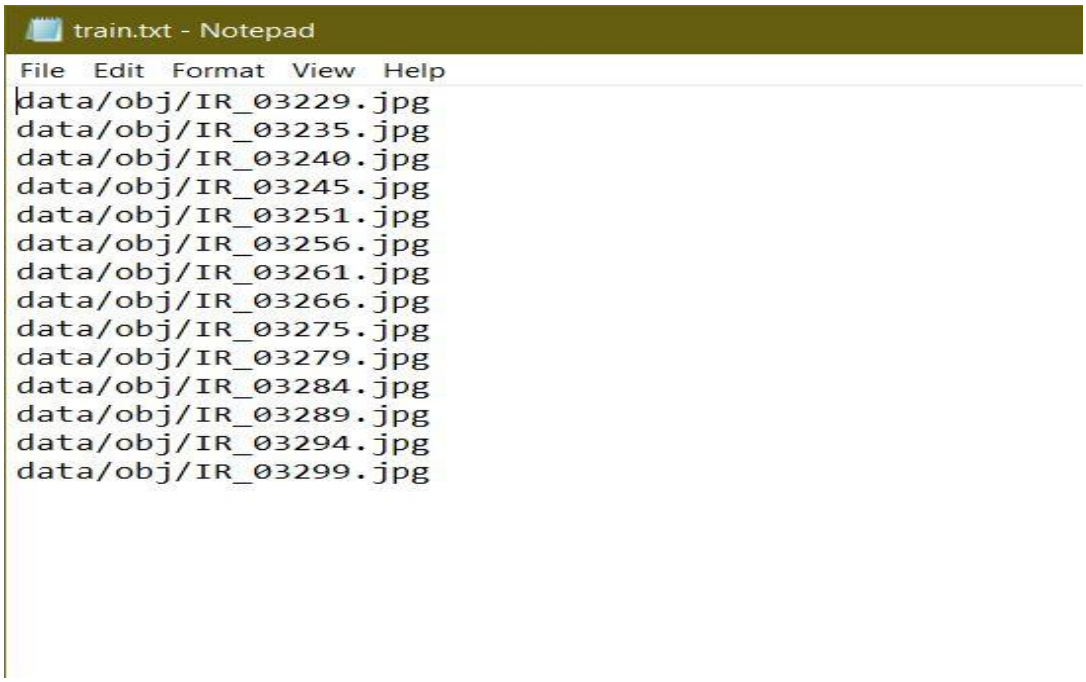


Fig 4. 15: Creating “train.txt” file to locate the training images

All the customization to train "YOLO" is done and only one thing is left is to provide the training images. A block diagram representation of "YOLO" customization is shown in Fig 4.16.

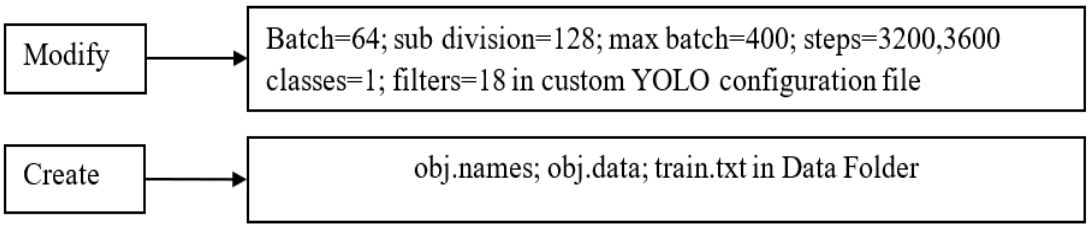


Fig 4. 16: Block diagram representation of customization in YOLO configuration file

4.4.4 Processing Training Images For YOLO

The last thing of experimental setup is to process all the training images before giving to "YOLO" for training. First of all, images taken by infrared camera Tis10 are saved in ".IS2" format which is not accessible for further use. A proprietary software known as "SmartView" is used to convert ".IS2" file to ".jpg" file. At first, "SmartView" software is opened and import the raw image file. Inside this software different parameters can be edited. For example: highest temperature markers on PV module as shown in Fig 4.17. After necessary adjustment, the raw image file is exported to a new folder in ".jpg" format.

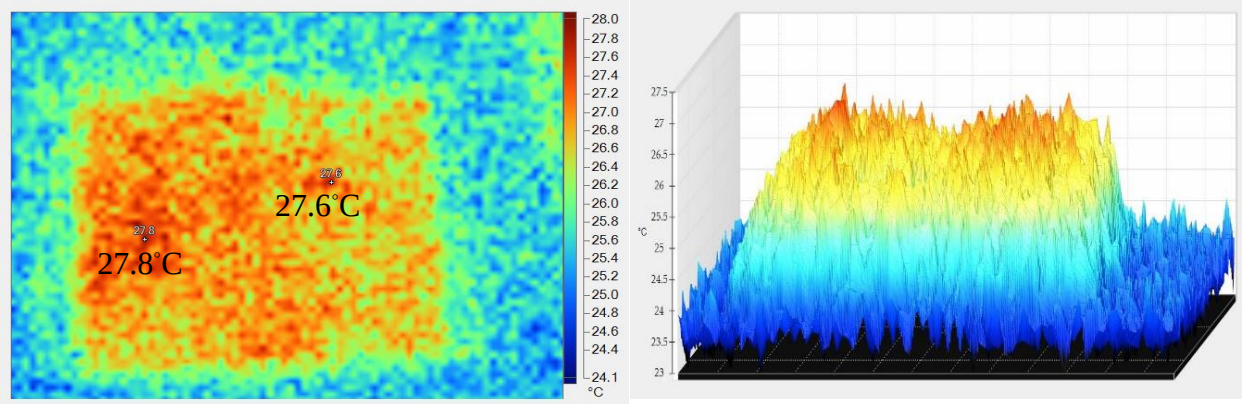


Fig 4. 17: a) Highest temperatures on PV module's surface and b) 3D temperature representation of the PV module using "SmartView"

After extracting all the thermal image to ".jpg" format it's time to manually detect the probable hotspot region using python. To complete this step, "Anaconda" a python distribution needs to be installed in the windows pc. "Anaconda" is used to create suitable environment and helps to install all necessary libraries needed to do image processing by python. Fig 4.18 shows the graphical interface of "Anaconda" where environment was set for python version 3.6. After creating the environment, a particular python integrated development environment (IDE) is selected that is "Jupyter Notebook". This is a web-based application where all the codes of python are compiled. Before staring the code, few libraries like CV2, NumPy, matplotlib need to import which will help to run the program. After compiling the python program, a contour of probable hotspot is presented as output shown in Fig 4.19

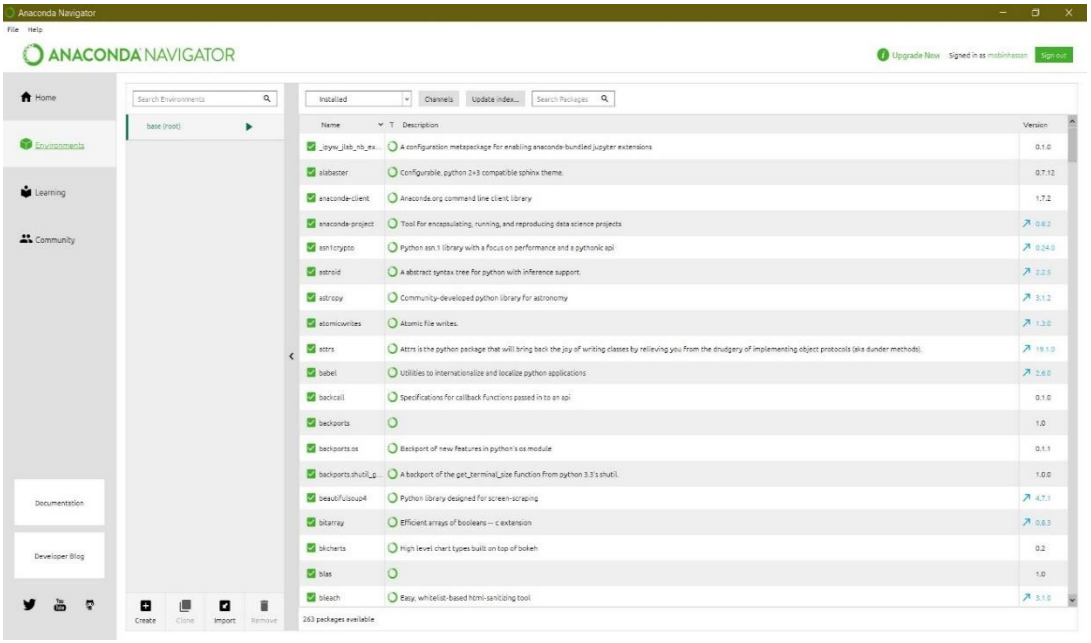


Fig 4. 18: Graphical interface of anaconda distribution

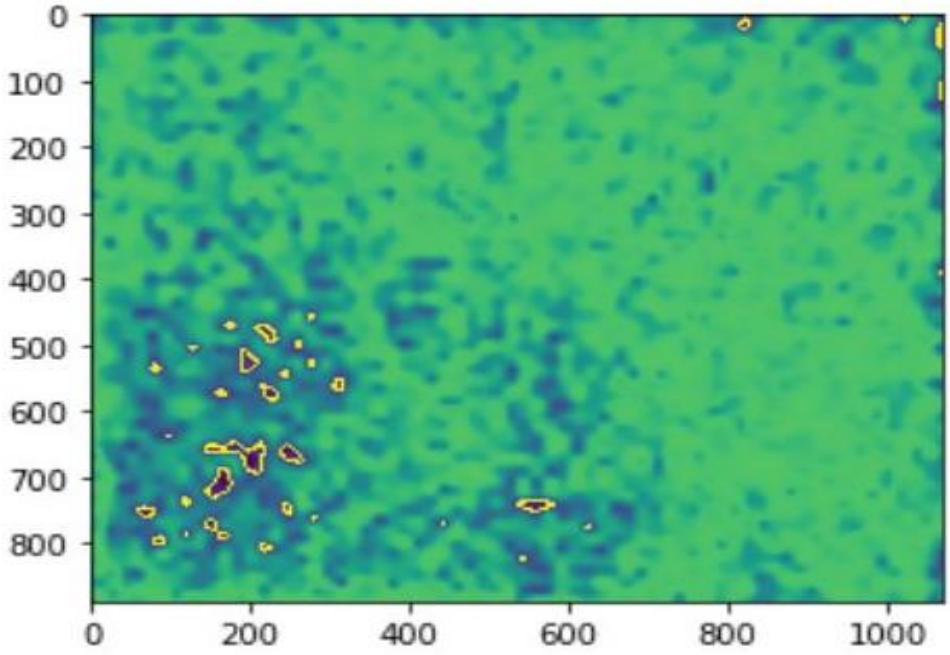


Fig 4. 19: Contour detection of the hotspots from the thermal image of PV module

The next step is to label all the contour region so that it can be used as training data set for machine learning “YOLO”. To label image for YOLO a particular software is used known as “LabelIMG”. At first, a thermal image is imported to “LabelIMG” and label all the hotspot regions manually by comparing the corresponding processed image in python side by side as shown in Fig 4.20. “LabelIMG” will convert “.jpg” file to “.txt” after labeling. In the “.txt” file there are five column and the number of rows depend on the total labeled object on the image as shown in Fig 4.21. The column defines the class of the image and in this study only one class is present which is “Hotspot” and that is why it is “0” in the first column. This means, the algorithm will only detect hotspots on any given image. And other columns represent the position of each object labeled.

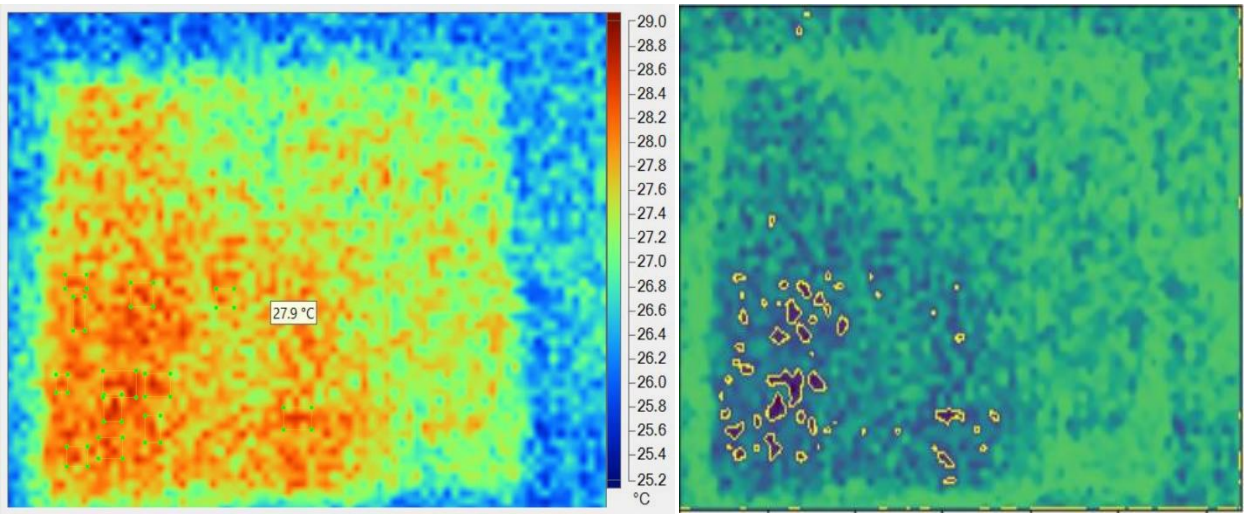


Fig 4. 20: Labelling training image manually by comparing with the processed image

All the text file of each training images is saved in the "obj" folder inside "Data" folder of darknet with their respected thermal images and name of text file and the image file must be same. After this all the steps of software setup are complete and the detector is ready for training.

IR_03229.txt - Notepad

File Edit Format View Help

```
0 0.105469 0.840292 0.031250 0.031315
0 0.176563 0.786534 0.015625 0.049061
0 0.191016 0.735908 0.060156 0.031315
0 0.483203 0.813152 0.039844 0.025052
0 0.114062 0.599165 0.015625 0.064718
0 0.196875 0.603340 0.046875 0.043841
0 0.224219 0.556889 0.029687 0.044885
0 0.248047 0.743737 0.033594 0.042797
0 0.293359 0.626305 0.028906 0.058455
0 0.170703 0.871086 0.030469 0.040710
0 0.169922 0.430063 0.027344 0.037578
0 0.187500 0.555324 0.025000 0.027140
```

Fig 4. 21: Text file of training image which issued in training

4.5 Summary

In the end, this chapter is divided into two setups, one is Hardware setup and other is Software setup. In hardware part, all the steps starting from taking thermal image and measuring IV data are stated. On the other, in software part all the environment, software, parameters are configured to run “YOLO”.

In the hardware part the main problem was to keep the temperature of the wooden box around 25°C. And in software section few problems like, matching equivalent software with the hardware of the PC, different errors due to mismatch directory path.

Chapter 5:

Result Analysis and Discussion of IV of the PV Modules

5.1 Introduction

In this chapter I-V profile of modules are analyzed. There are total fifteen modules of three different companies ("Generic", "Greenland", "Solar"). So, all the PV modules are divided into three groups according to their manufactured company. After analyzing the I-V profile of three groups and their IR images, a relation between the hotspot and defected area is established. Further, reason for deviation in I-V graph in few modules is speculated.

5.2 I-V data of PV modules

There are total fifteen PV modules of three different companies like: Generic, Greenland and Solar. Five similar PV modules from each company were taken for the experiment and different parameters of the modules are shown in the tables below

Table 5. 1: Experimental data of "Generic" company's five modules

Parameters	Pmax	Voc	Isc	ΔP
Module 1	4.814W	21.17V	0.312A	0%
Module 2	4.709W	21.52V	0.297A	2.18%
Module 3	4.427W	20.81V	0.302A	8.04%
Module 4	4.389W	21.2V	0.303A	8.83%
Module 5	4.158W	18.81V	0.299A	13.63%

Table 5. 2: Experimental Data of "Greenland" company's Five Module

Parameters	Pmax	Voc	Isc	ΔP
Module 6	4.896W	21.44V	0.307A	0%
Module 7	4.896W	21.45V	0.307A	0%
Module 8	4.731W	21.43V	0.3A	3.37%
Module 9	4.587W	21.45V	0.299A	6.31%
Module 10	4.413W	21.56V	0.319A	9.87%

Table 5. 3: Experimental Data of "Solar" company's Five Module

Parameters	Pmax	Voc	Isc	ΔP
Module 11	5.407W	19.54V	0.387A	0%
Module 12	4.911W	19.04V	0.375A	9.17%
Module 13	4.71W	19.5V	0.371A	2.89%
Module 14	4.306W	18.8V	0.337A	20.36%
Module 15	4.156W	18.9V	0.35A	23%

Above, three different tables are created based on the companies. Table 5.1, Table 5.2, Table 5.3 shows all the parameters of "Generic", "Greenland" and "solar" company's modules respectively. Inside each table, there are four different parameters and five sample PV modules. The parameters are, Maximum power (Pmax), Open-circuit Voltage (Voc), Short-circuit Current (Isc) which are obtain from I-V experiment. From each group one module is taken as standard which has maximum power. "module 1", "module 6" and "module 11" have shown highest power among the five-sample modules of "Generic", "Greenland" and "Solar" respectively. The

last parameter that is Performance Degradation (ΔP) is inserted by comparing the power of each module with the standard module of the individual group.

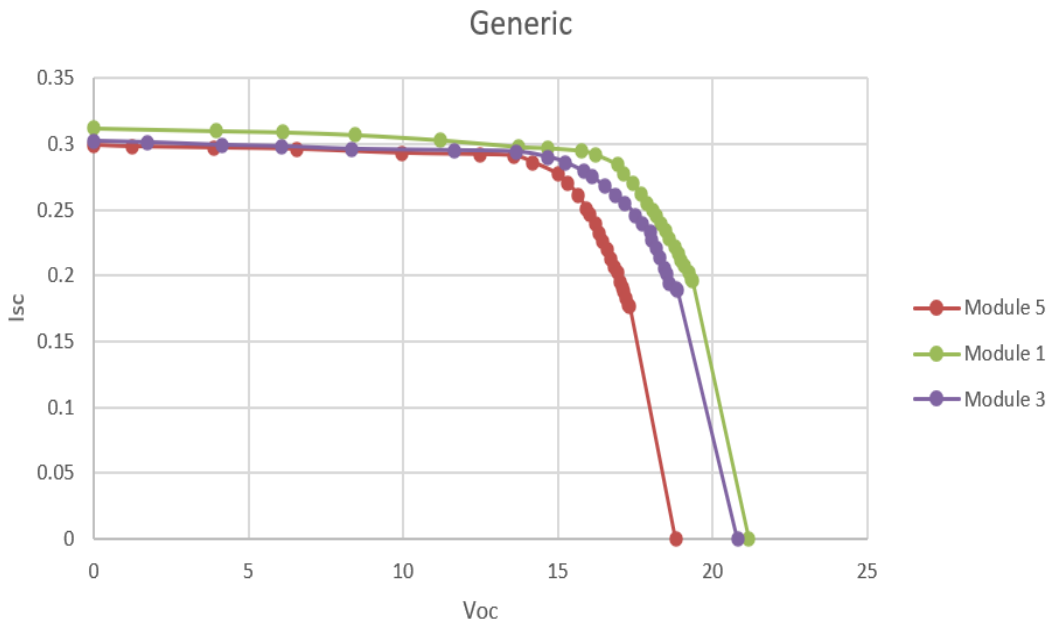


Fig 5. 1: I-V Characteristics of "Generic" modules measured at 25°C and under 500 W/m^2 irradiation level

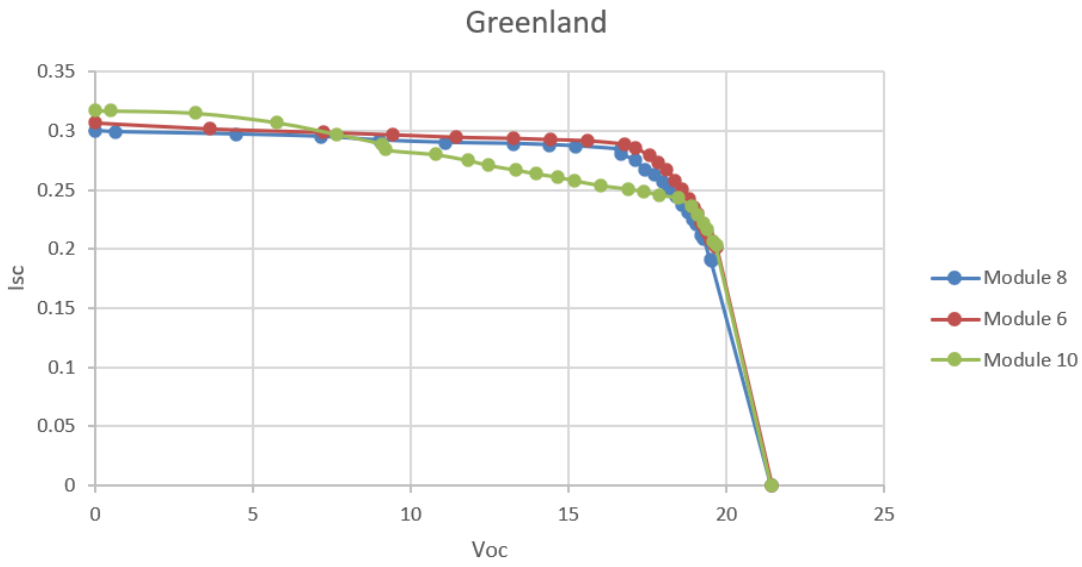


Fig 5. 2: I-V Characteristics of "Greenland" modules measured at 25°C and under 500 W/m^2 irradiation level

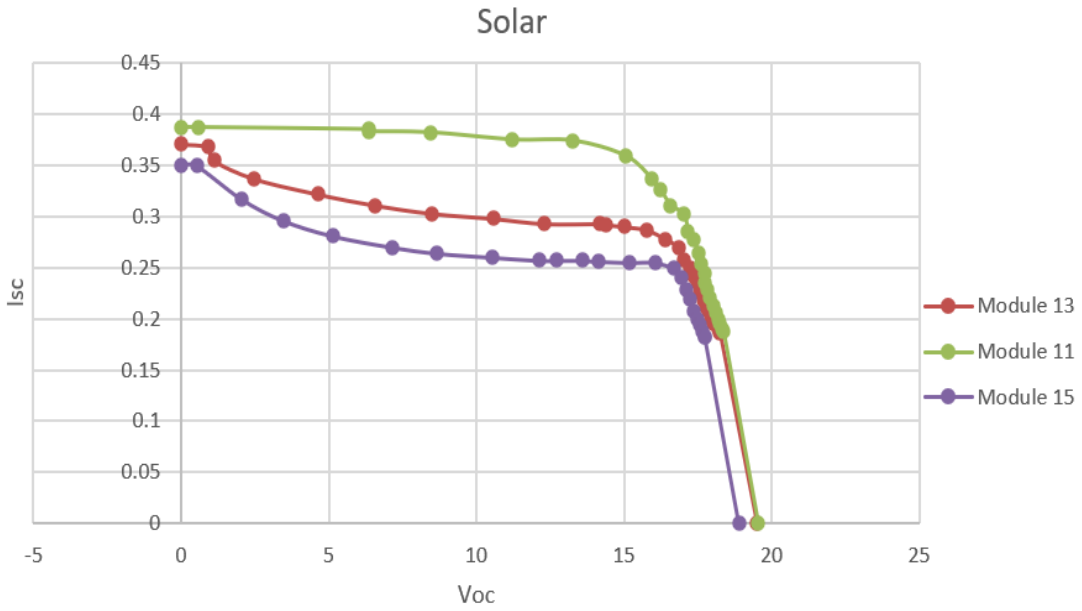


Fig 5. 3: I-V Characteristics of "Solar" modules measured at 25°C and under 500 W/m^2 irradiation level

Fig 5.1, Fig 5.2 and Fig 5.3 shows the combine I-V curve of each group. All modules were experimented in same experimental setup that is 25°C and 500 W/m^2 irradiation. Here it is

clearly seen that, all the modules of “Generic” follows normal I-V curve and module (“module 10”) of “Greenland” slightly bends from the standard I-V curve. So, it can be said that in these two modules shunt resistance is less compare to the other modules of this group. But, modules (“module 13”, “module 15”) of “solar” do not follow the standard I-V curve. All the three modules have very less shunt resistance and there is rise of Isc near the vertical axis because of bypass diode short circuit.

5.3 Infrared images of modules

In this section, all the data collected from IR image of the PV modules are analyzed. When each PV modules are supplied with their respective rated Voc and Isc for one minute, there is a temperature change on the surface of the module with respective to the ambient temperature. This temperature change is calculated with the help of IR image and a temperature sensor DHT11. In this study, two different temperature differences are calculated. One is between highest temperature and ambient temperature; another is between average temperature of the surrounded area of that highest temperature point and ambient temperature.

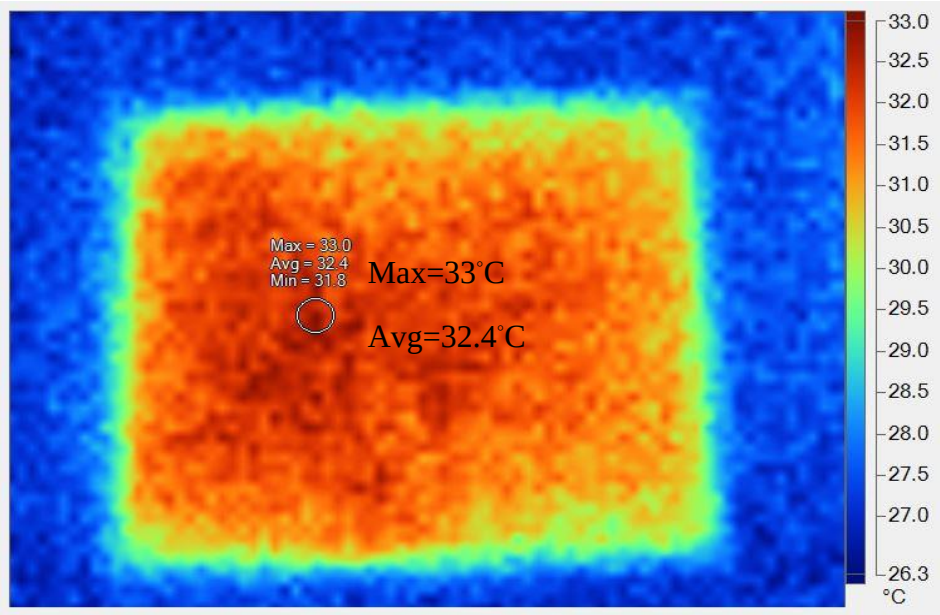


Fig 5. 4: Average temperature of the surrounding area of the highest temperature point of the PV module shown as “Avg”

Table 5. 4: Temperature difference of "Generic" modules with respect to ambient temperature in one-minute time frame

Parameters	Max Temperature	Avg Temperature	Ambient Temperature	ΔT (Max Temperature)	ΔT' (Avg Temperature)
Module 1	28.7	27.94	26.1	2.6°C	1.84°C
Module 2	28.6	27.89	25.8	2.8°C	2.09°C
Module 3	30.2	29.45	26.3	3.9°C	3.15°C
Module 4	30.9	29.73	25.9	5°C	3.83°C
Module 5	31	30.45	26.1	4.9°C	4.35°C

Table 5. 5: Temperature difference of "Greenland" modules with respect to ambient temperature in one minute time frame

Parameters	Max Temperature	Avg Temperature	Ambient Temperature	ΔT (Max Temperature)	ΔT' (Avg Temperature)
Module 6	30.1	29.32	25.7	4.4°C	3.62°C
Module 7	28.28	28.17	26.1	2.7°C	2.07°C
Module 8	28	27.29	26	2°C	1.29°C
Module 9	29.7	29.04	25.7	4°C	3.34°C
Module 10	28.2	27.58	26.1	2.1°C	1.48°C

Table 5. 6: Temperature difference of "Solar" modules with respect to ambient temperature in one minute time frame

Parameters	Max Temperature	Avg Temperature	Ambient Temperature	ΔT (Max Temperature)	ΔT' (Avg Temperature)
Module 11	28.1	27.31	25.5	2.6°C	1.81°C
Module 12	28.7	27.94	25.8	2.9°C	2.14°C
Module 13	31.6	30.99	25.7	5.9°C	5.29°C
Module 14	29.1	28.52	26.3	2.8°C	2.22°C
Module 15	26.7	26.10	23	3.7°C	3.10°C

Maximum Temperature difference (ΔT) is calculated from the differnce between highest value of the module (which can be found from 3D surface temperature diagram of each module as shown in Fig 5.5) and ambient temperature. And average temperature (ΔT') is calculated from the diffrence between average temperature of the surronding area of the highest point (which can be found from"SmartView" output as shown in Fig 5.4) and the ambient temperature.

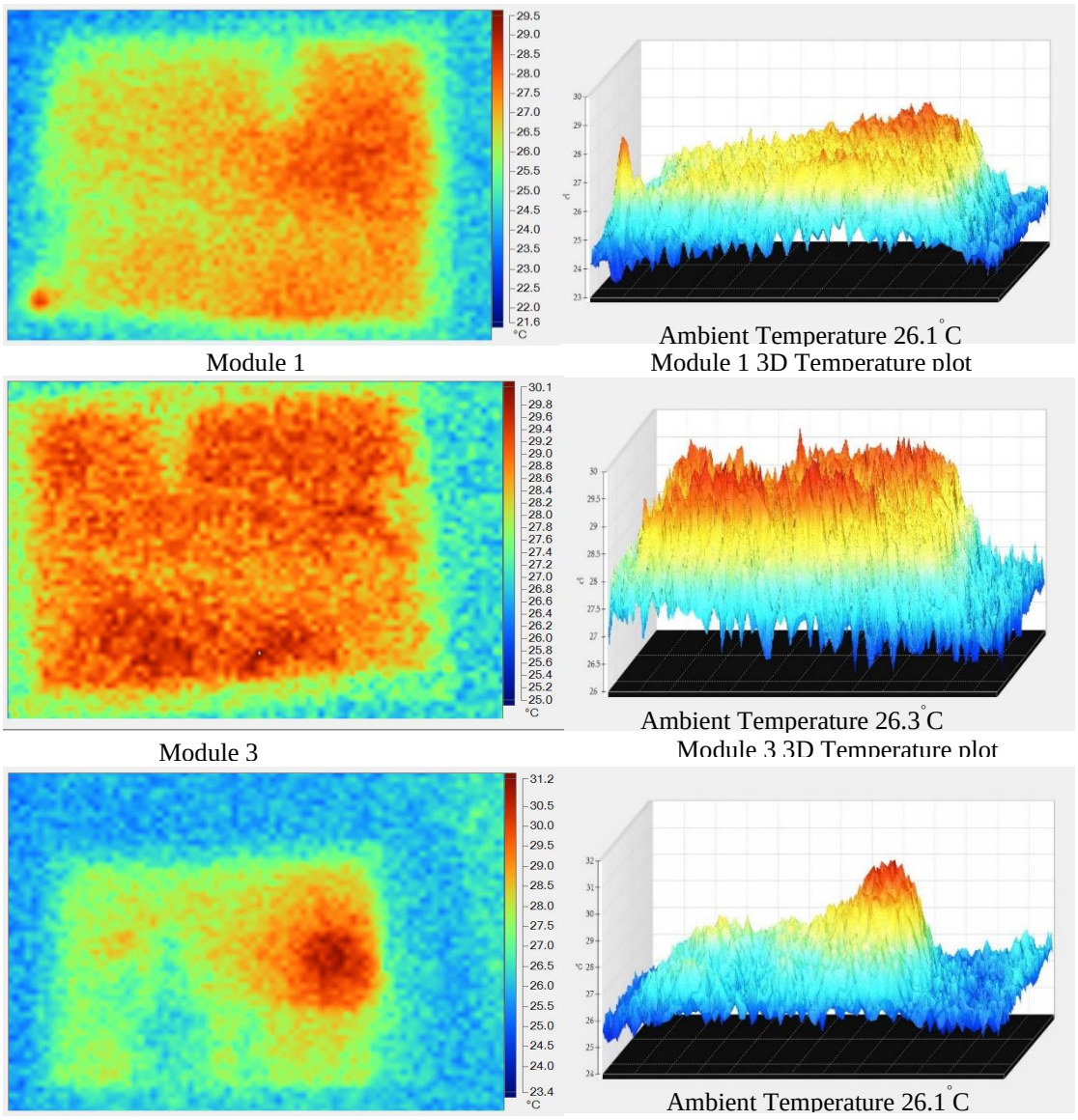
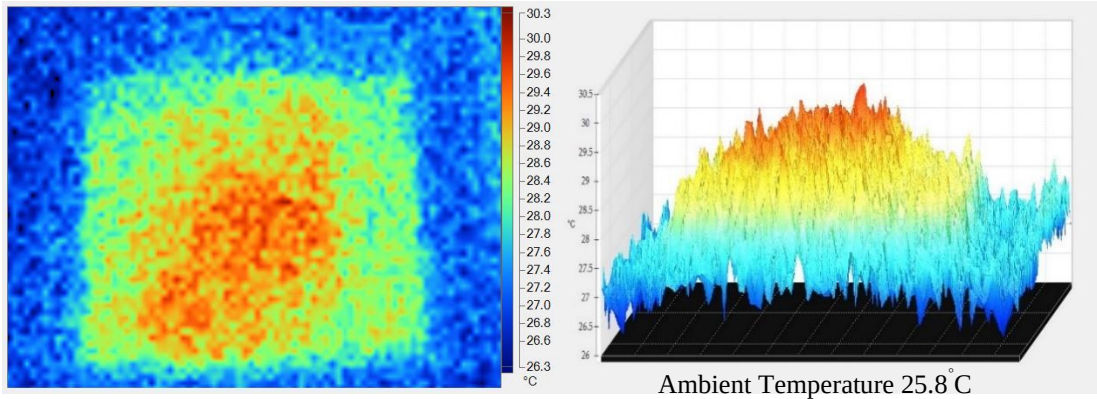
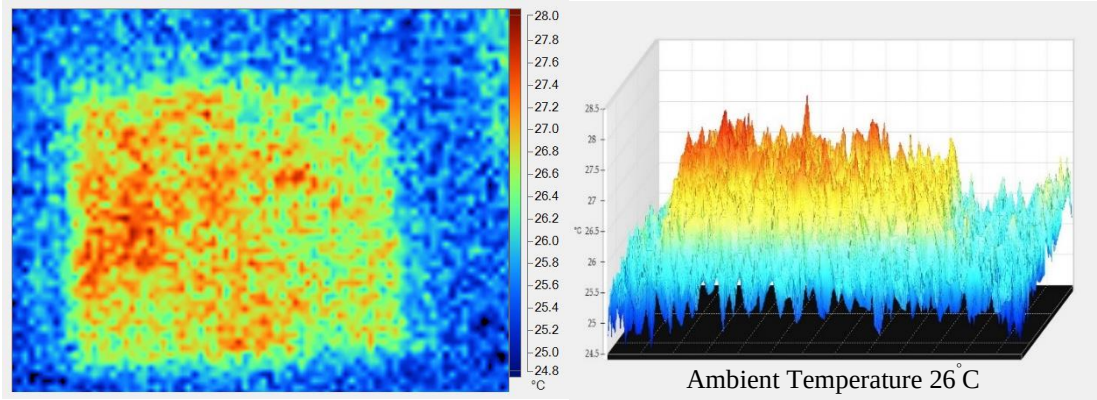


Fig 5. 5: Sample IR image of PV modules in one-minute time frame from Generic company with their respective 3D temperature diagram



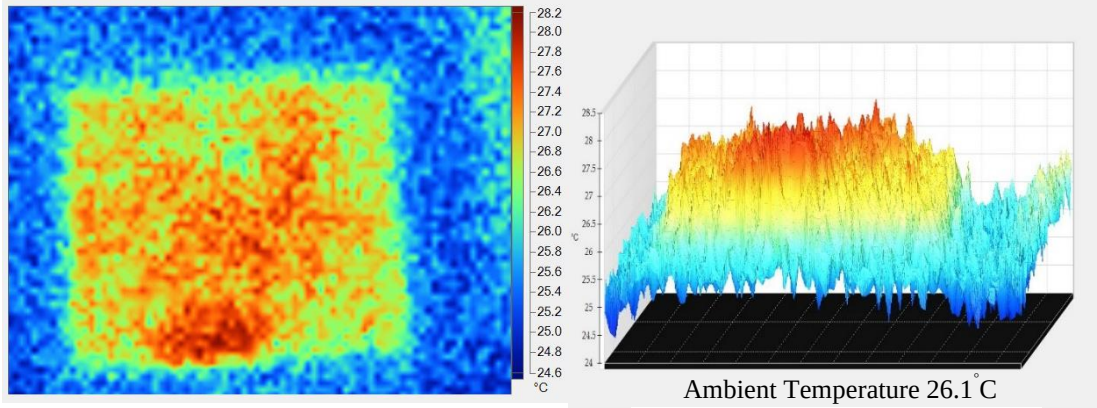
Module 6

Ambient Temperature 25.8 °C
Module 6 3D Temperature plot



Module 8

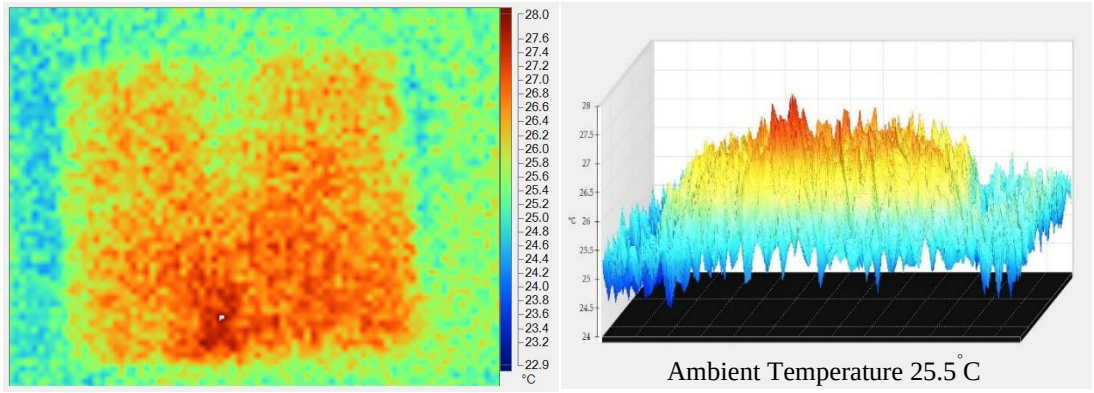
Ambient Temperature 26 °C
Module 8 3D Temperature plot



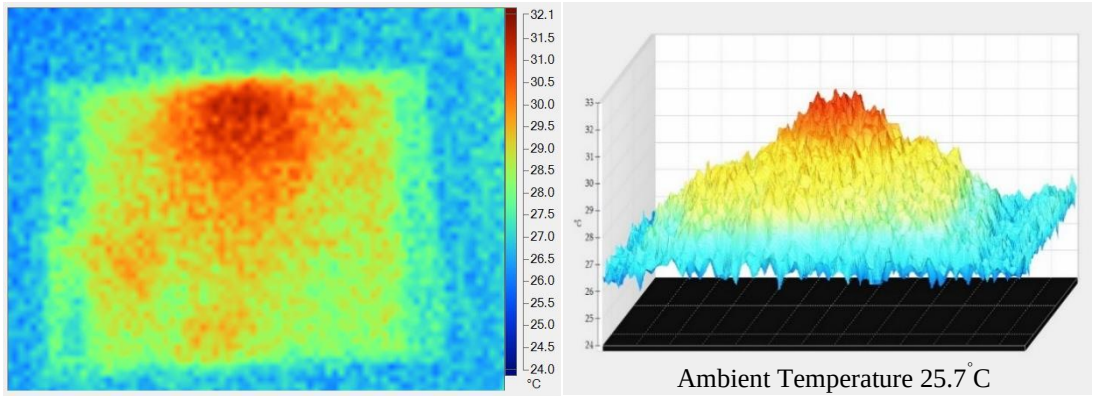
Module 10

Ambient Temperature 26.1 °C
Module 10 3D Temperature plot

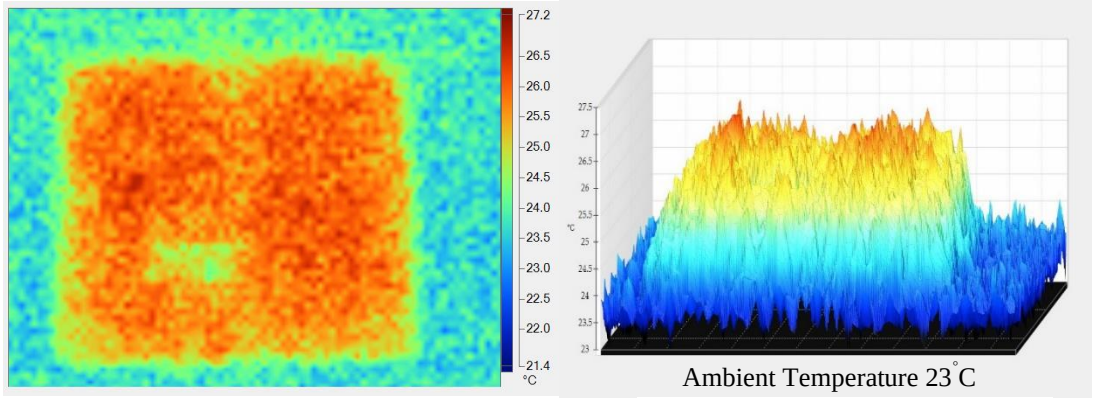
Fig 5. 6: Sample IR image of PV modules in one-minute time frame from Greenland company with their respective 3D temperature diagram.



Module 11 | Module 11 3D Temperature plot



Module 13 | Module 13 3D Temperature plot



Module 15 | Module 15 3D Temperature plot

Fig 5. 7: Sample IR image of PV modules in one-minute time frame from Solar company with their respective 3D temperature diagram.

5.4 Combine Data Analysis

All the modules were run to their maximum power by supplying rated current and voltage for twenty minutes. As the time passes temperature of the module's surface increases. Temperature of modules increases due to the impact of shunt and series resistance. So, it can be said that more temperature rise means more shunt and series resistance impact and due to high temperature hotspots are created which can be seen using thermal camera.

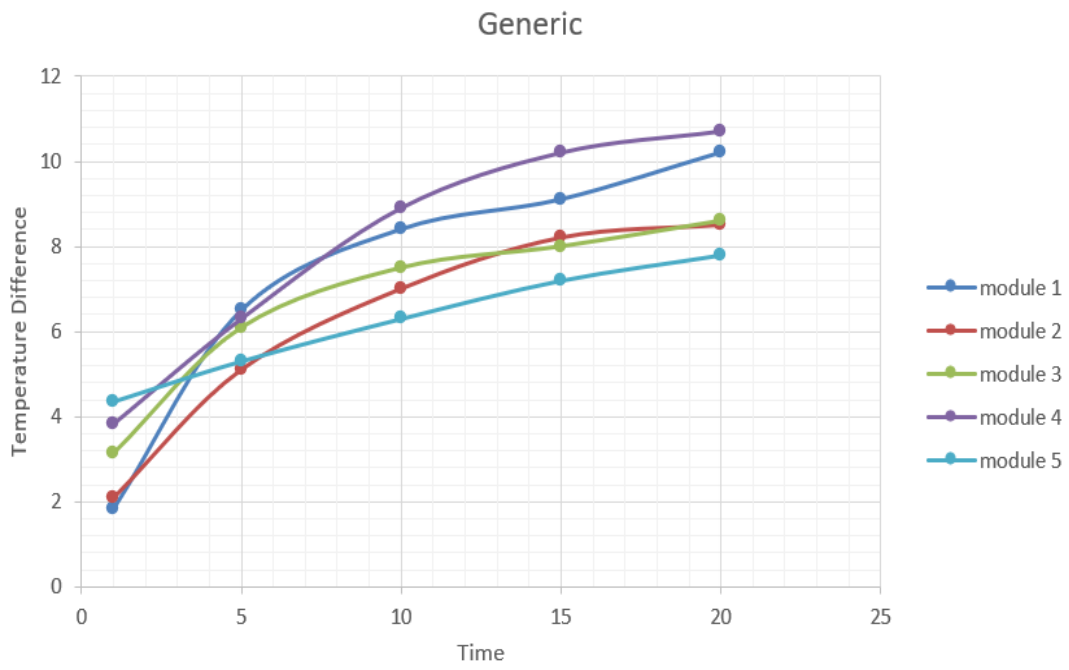


Fig 5. 8: Change of average temperature difference of "Generic" modules with respect of time

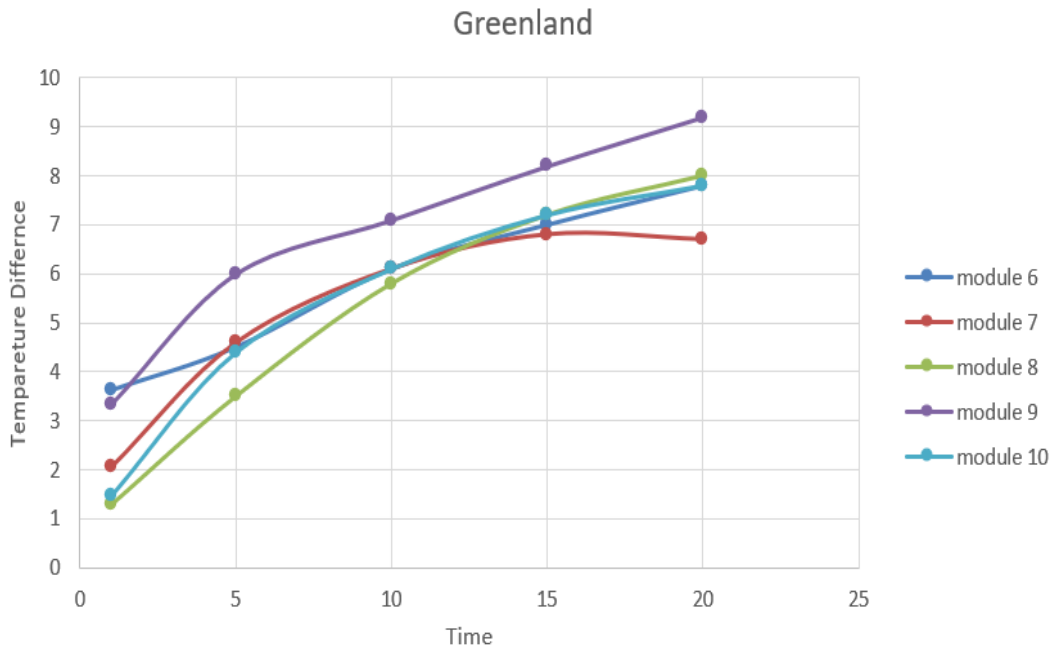


Fig 5. 9: Change of average temperature difference of “Greenland” modules with respect of

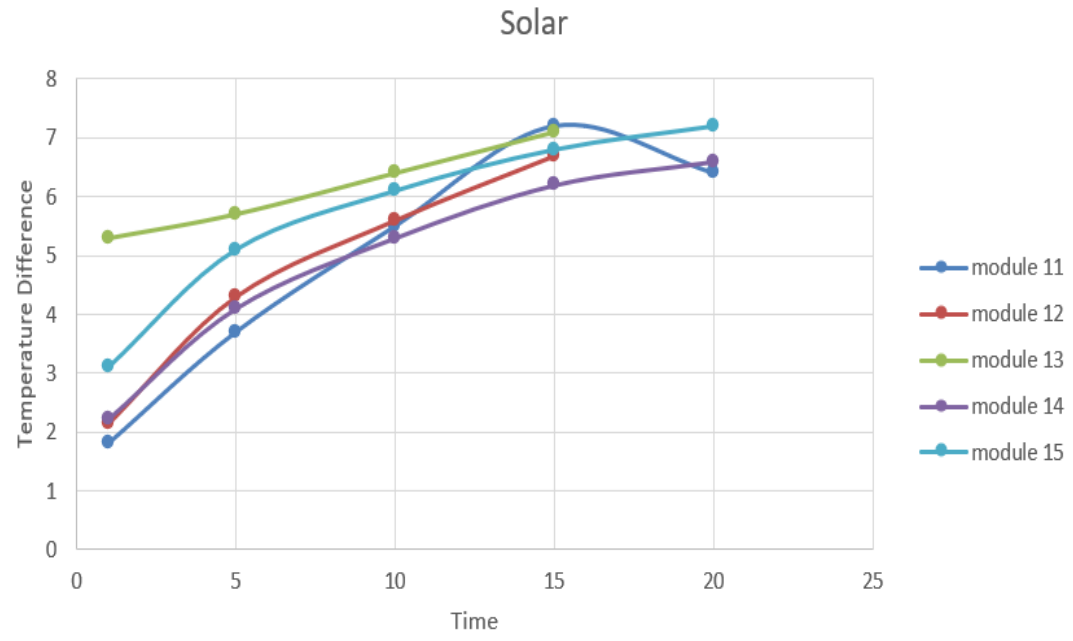


Fig 5. 10: Change of average temperature difference of “Solar” modules with respect of time

From the Fig 5.8 it is seen that “module 4” started with an initial average temperature difference of 3.83°C and after twenty minutes the average temperature difference rises to 10.7°C and on the other hand “module 5” of the same group started with 4.35°C and after twenty minutes it rises to 7.8°C. so it can be said that there is more heat loss in “module 5” compare to “module 4” and defected area should be higher in “module 5”. But defected area of a module does not have any relation with average temperature difference as shown in Fig 5.11. All data of defected area and average temperature of each module is shown in Table 5.7

Table 5. 7: Defected area and average temperature difference of all fifteen modules.

Module No by Company	Defected Area (%)	Avg Temperature Difference (ΔT’)
Module 1	2.3	1.84
Module 2	6.362	2.09
Module 3	4.824	3.15
Module 4	0.916	3.827
Module 5	2.523	4.35
Module 6	1.016	3.62
Module 7	1.815	2.07
Module 8	1.361	1.29
Module 9	0.106	3.34
Module 10	4.218	1.48
Module 11	1.475	1.81
Module 12	0.075	2.14
Module 13	2.066	5.29
Module 14	2.253	2.22
Module 15	0.572	3.103

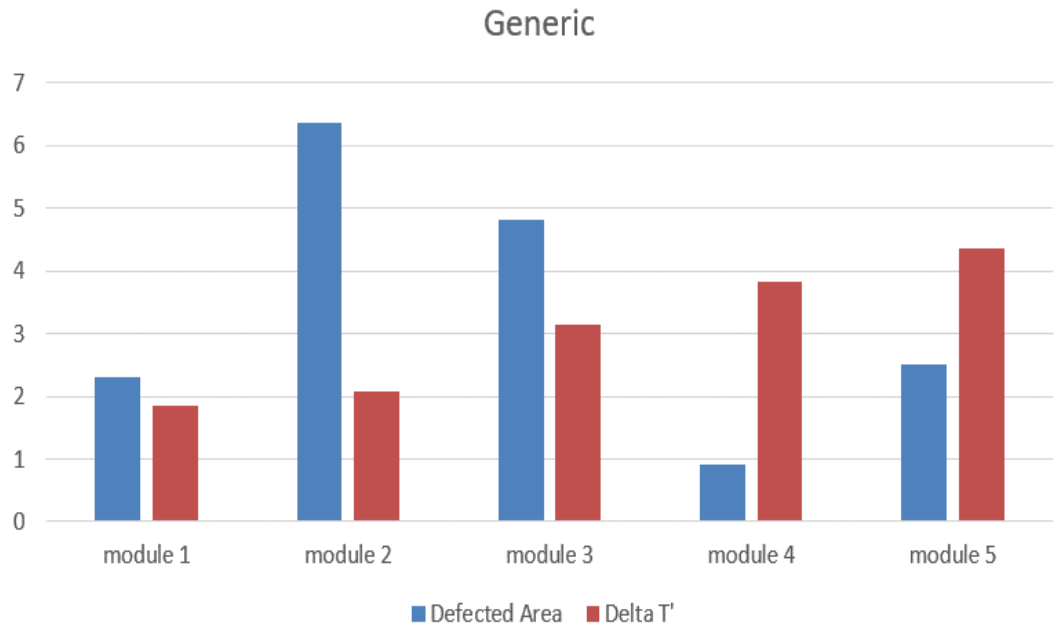


Fig 5. 11: Comparison between defected area and average temperature difference of Generic modules

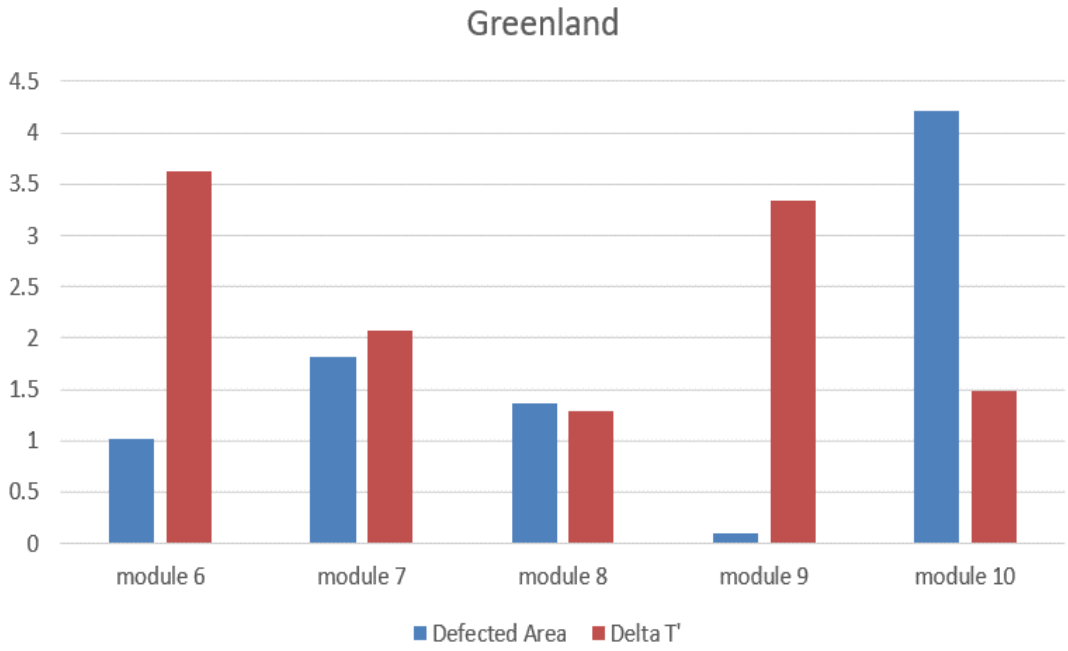


Fig 5. 12: Comparison between defected area and average temperature difference of "Greenland" modules

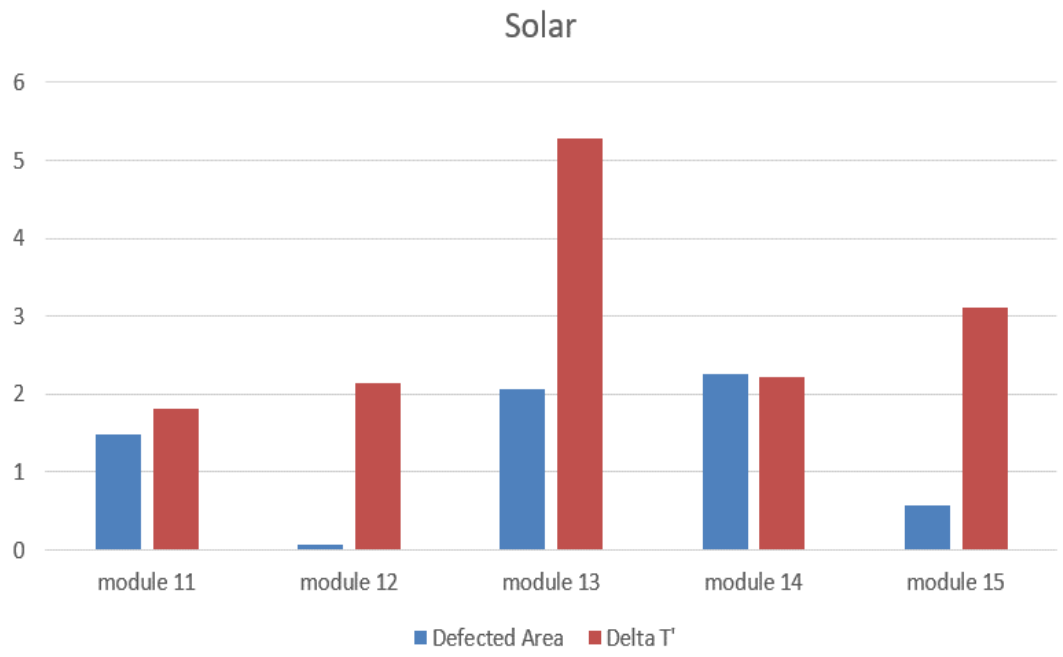


Fig 5. 13: Comparison between defected area and average temperature difference of Solar modules

From the above chart between defected area and Average temperature difference there is no interconnection. A PV module can have less defected area but high average temperature difference as it can be seen in case of "module 4" in Fig 5.11 has less defected area but high average temperature difference compares with the other modules in the same group. On the other hand, PV module can have high defected area but less average temperature difference as it can be seen in "module 10" in Fig 5.12. it has high defected area but less average temperature difference compares with other modules of the same group.

5.5 Summary

In this chapter, I-V characteristics of each PV module is analyzed. Few modules don't follow the I-V graph because of series and shunt resistance. Hotspot area in a solar panel does not have any relation with the average temperature difference which is shown in a chart.

Chapter 6:

Result analysis of “YOLO” detector

6.1 Introduction

In this chapter, the outcome of YOLO detector will be analyzed. In this study, YOLO detector was trained two times. In first case, only ten images were given for training and in second case, total fourteen images were given. The purpose was to see the difference whether the detector improves itself with more data or not. Next, the final trained detector (total fourteen images) was tested with various test images to validate its performance.

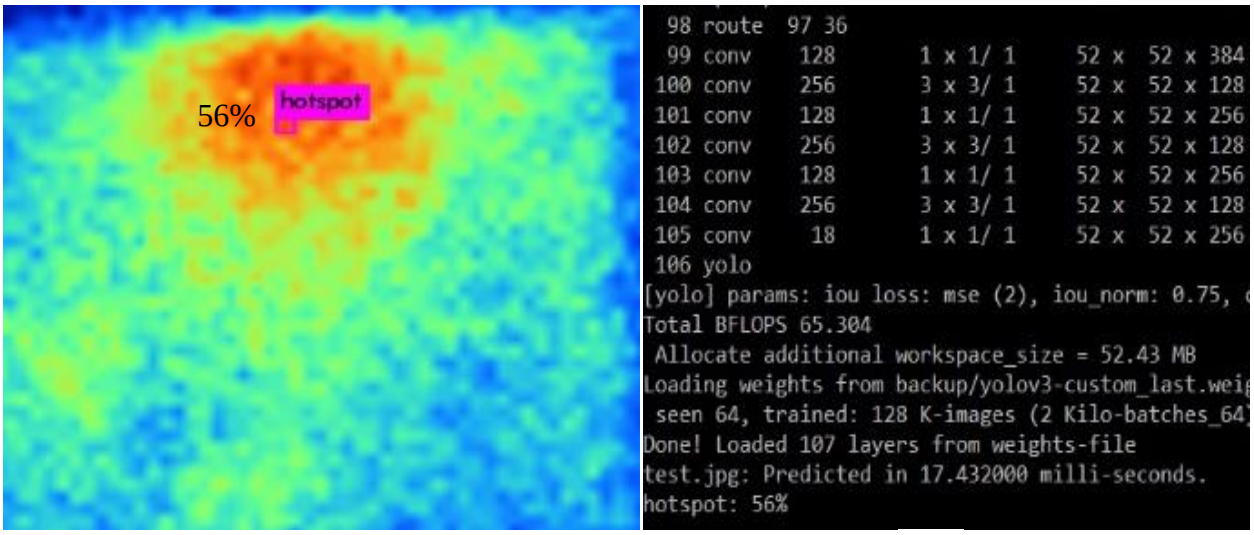
6.2 Performance of YOLO detector-1 with ten images for training

For training with ten images, thermal image of module- “1-10” were selected. And module- 13 was set apart in both the training. In case of ten images, training was set to undergo 4000 iterations, where it will take a batch of images and keep training on until max iteration is reached. A loss vs Iteration graph was plotted, where it is observed that the loss starts to decrease from around 400 iteration as shown in Fig 6.1 and it took around one and half hours to train.

Fig 6. 1: Loss vs Iteration graph of ten images

An eye had to be kept on the loss as, if the loss becomes near close to zero, it might result in an over fit of the data. An overfit of the data means that the detector has “memorized” the location of the hotspots, but this wouldn’t be a problem in our situation, as the testing image wasn’t included in any of the data set. Typically, a loss value anything between 0<L<1 will result in a good detection ability.

Thermal image of module- 13 was used as the testing image. This image was excluded from all training data sets because it has the highest ΔT and ΔT’. Maximum temperature difference with respect to ambient temperature of module- 13 is ΔT=5.9°C and average temperature difference with respect to ambient temperature is ΔT’=5.29°C. The detector was run on the test image and the first trial was quit a success. The output result is shown in Fig 6.2.

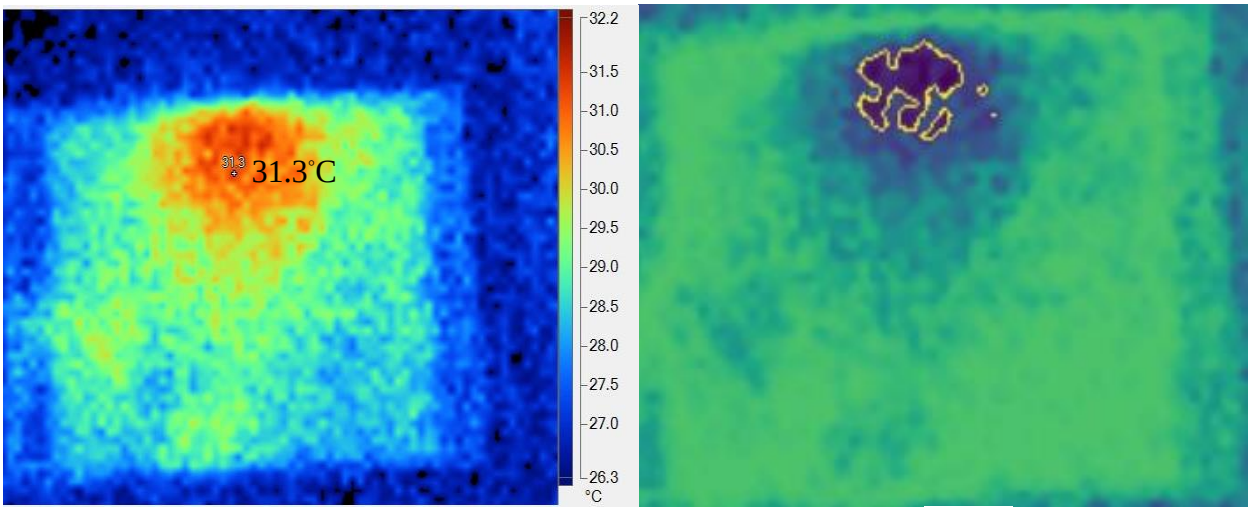


a b

Fig 6. 2: a) A hotspot is detected on module- 9 and b) the percentage of accuracy of the

It can be seen that the detector is able to identify a hotspot region, out of several that might exist in the given test image. The peak intensity value of the region identified by the detector is 31.3℃ and the actual hotspot of module -13 is shown is Fig 6.3. The ambient temperature of “module-13” in one-minute time frame was 25.7℃. so, the temperature difference is 5.6℃ and any spot

having more than 5℃ temperature deference with respect to ambient temperature can be said probable defected region. The detector is 56% confident about its detection as shown in Fig 6.2b.



a b

Fig 6. 3: a) Temperature of the detected region and b) actual hotspot of the module- 13

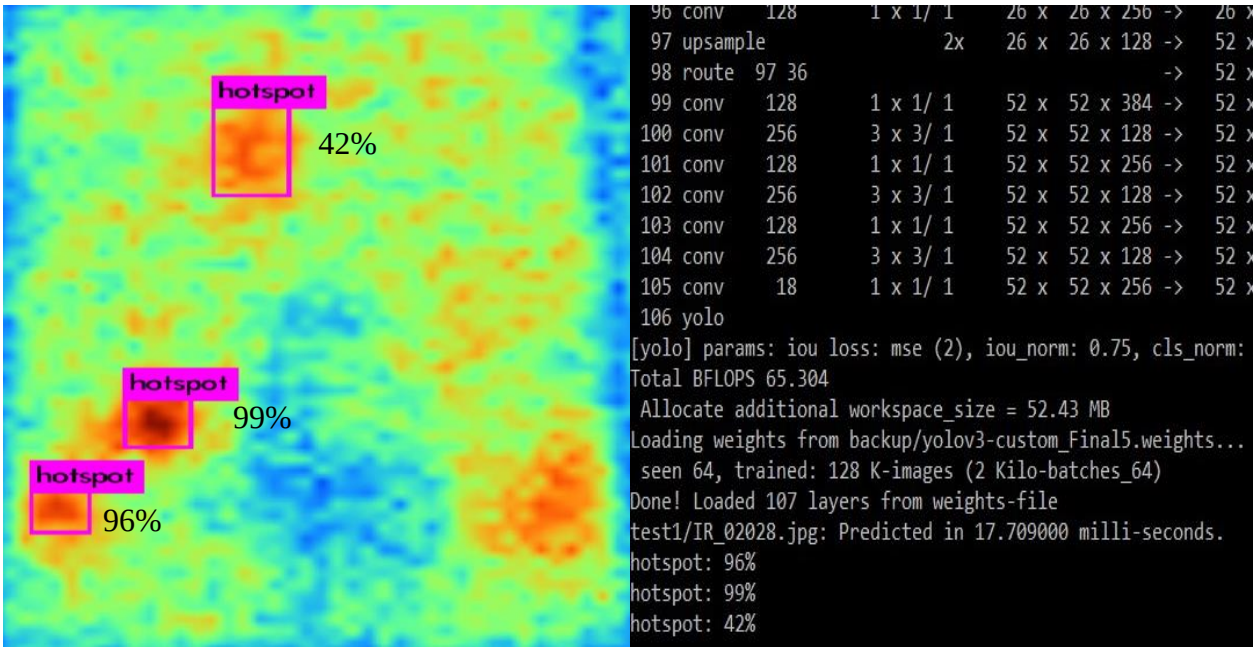
Another five test images were also tested with this particularly trained detector. In the following Figures results were shown.

Fig 6.4 shows the output of “Test image-1” and three detected regions. Among the three detection only one detection is correct as shown in Fig 6.5.

Fig 6.6 shows the output of “Test image-2” and three regions are detected and all the three detection are correct as shown in Fig 6.7. Yes, there are other hotspots on the PV module but it detected only three among all other.

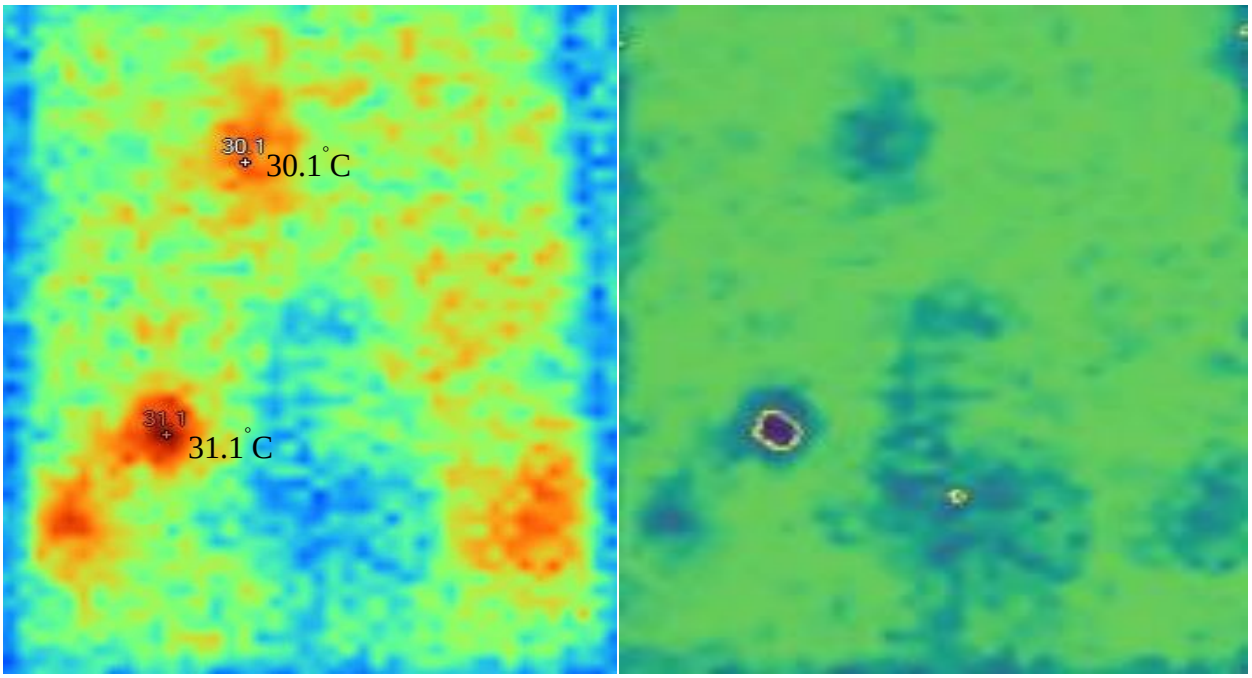
Again, Fig 6.8 shows the output of “Test image-3” and two regions are detected which are indeed hotspots as shown in Fig 6.9.

On the other hand, the detector failed to detect any hotspot in “Test image-4” and “Test image-5”.



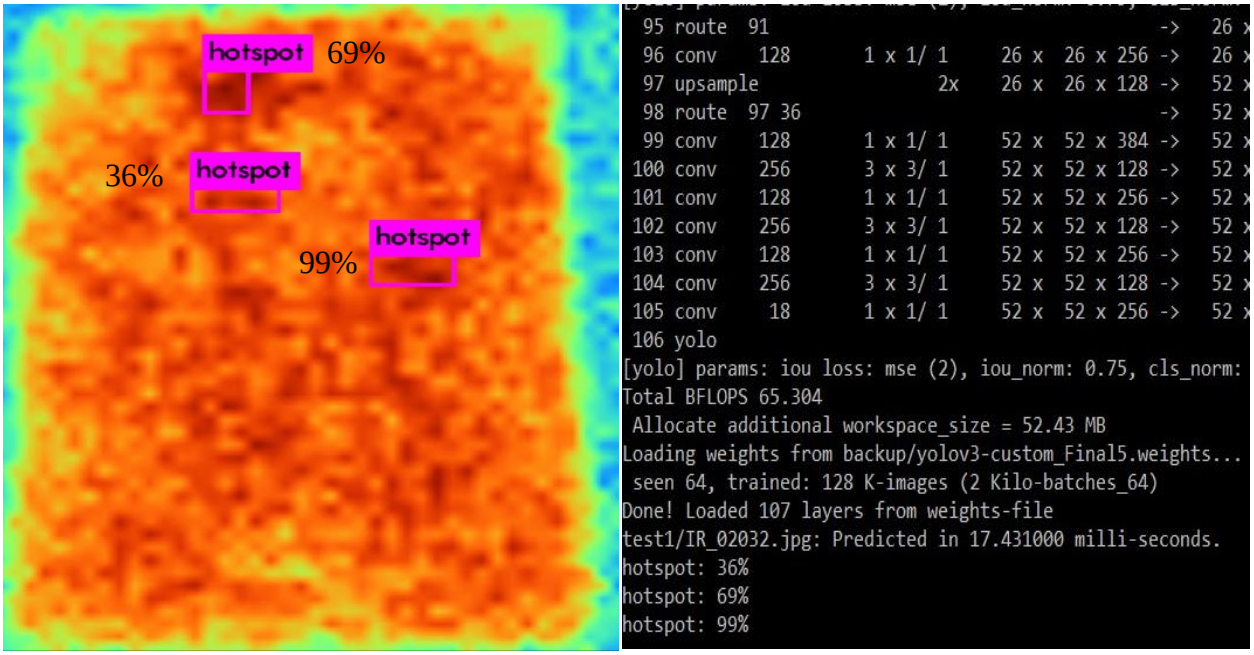
a b

Fig 6. 4: a) Three hotspots detected on Test image-1 and b) the percentage of accuracy of the hotspot



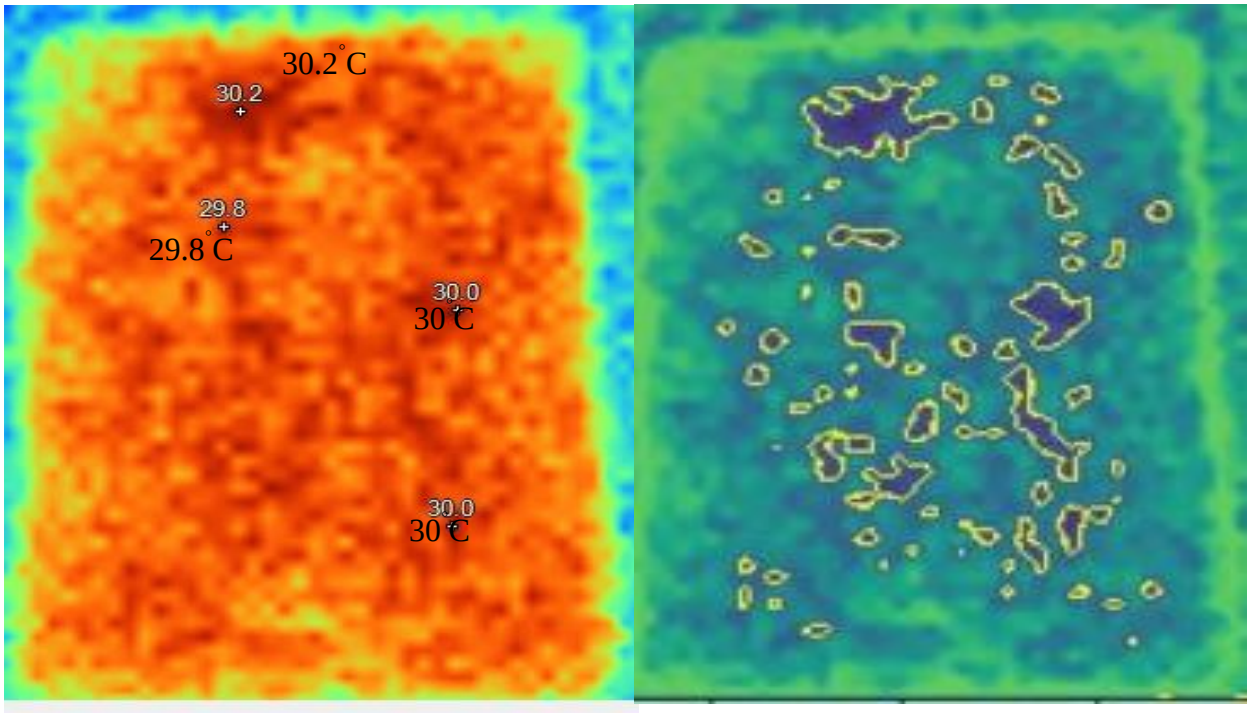
a b

Fig 6. 5: a) Temperature of the detected region and b) actual hotspot of the Test image-1



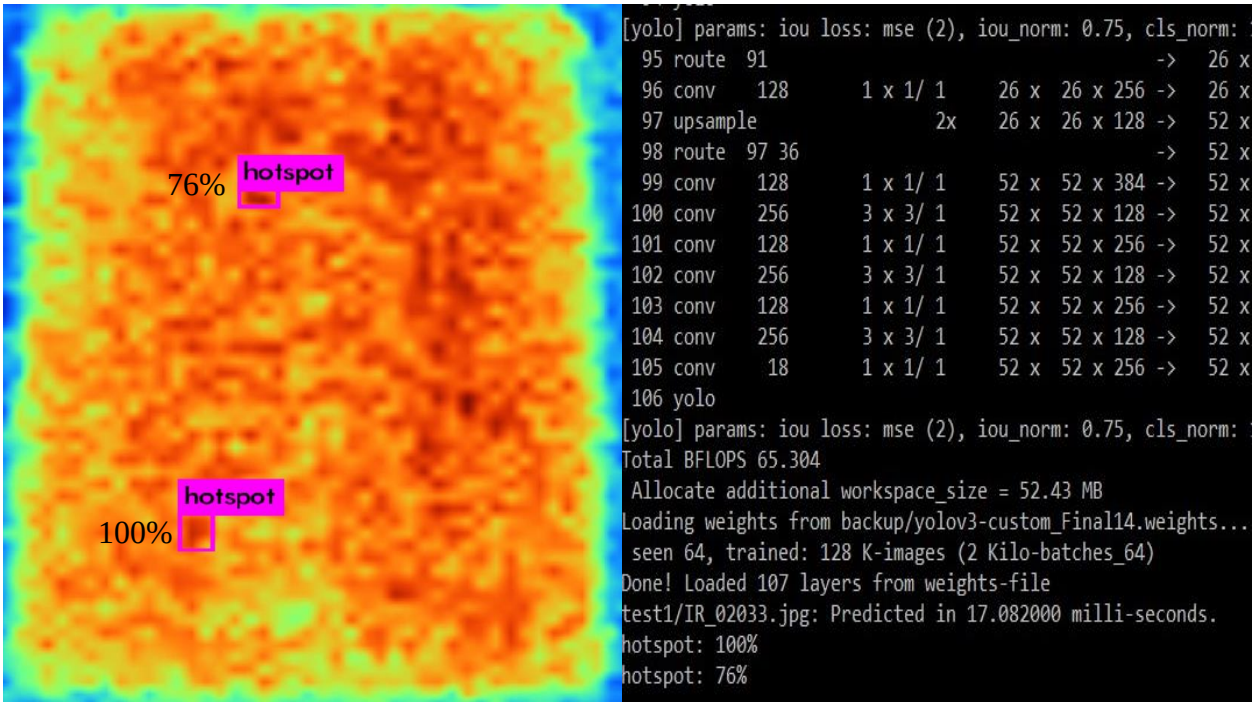
a b

Fig 6. 6: a) Three hotspots detected on Test image-2 and b) percentage of accuracy of the hotspots



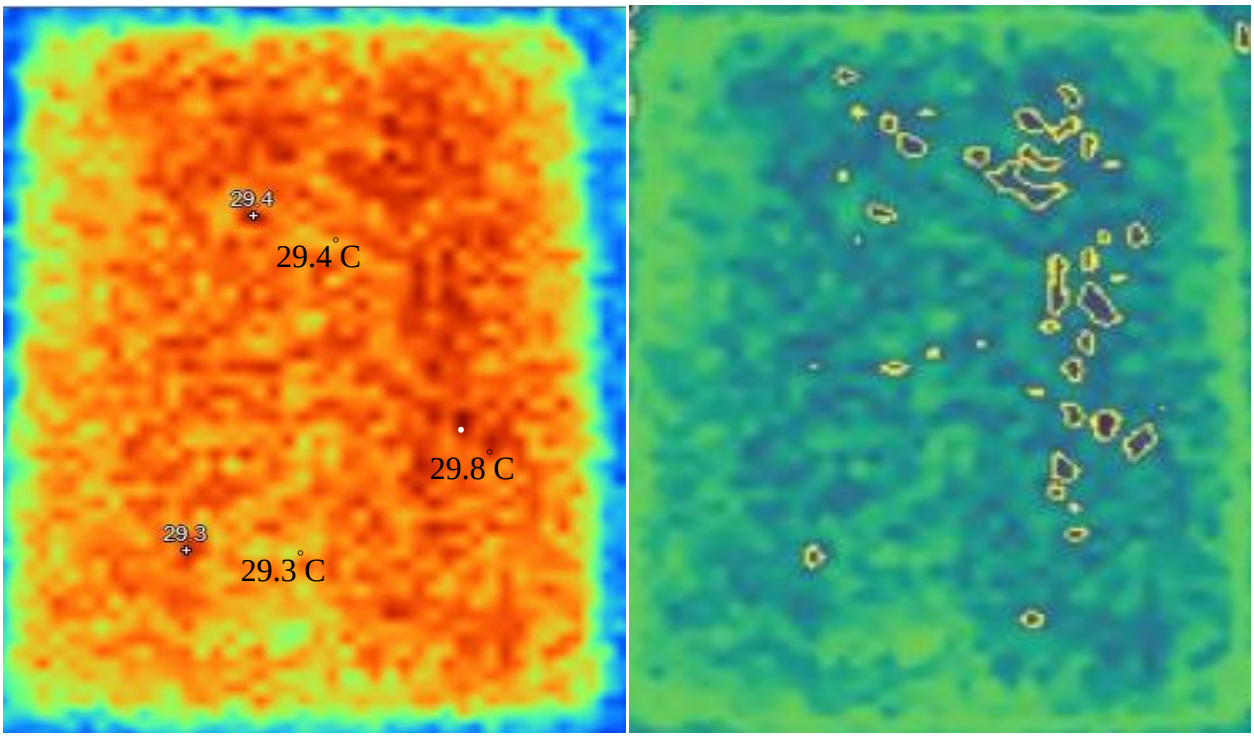
a b

Fig 6. 7: a) Temperature of the detected regions and b) actual hotspots of the Test image-2



a b

Fig 6. 8: a) Two hotspots detected on Test image-3 and b) percentage of accuracy of the



a b

Fig 6. 9: a) Temperature of the detected region and b) actual hotspots of the Test image-3

It is clear that the detector is able to detect hotspots quit accurately but it cannot yet detect all the probable hotspots and the reason is less training data. In the following table a summary of the detector-1 is shown.

Table 6. 1 Summarization of the detector-1 which is trained with Ten images

Test image	**Correct detection**	**Wrong detection**	**Confident on the highest temperature region**
Test image-1	1	2	99%
Test image-2	3	0	69%
Test image-3	2	0	Failed
Test image-4,5	0	0	Failed

6.3 Performance of YOLO detector-2 with fourteen images for training

In the final training, total fourteen images were given out of fifteen images to the detector for training. One image was kept aside which was the thermal image of module- 13 and reason is to compare the improvement. The loss vs iteration graph of fourteen image training is shown in Fig 6.10.

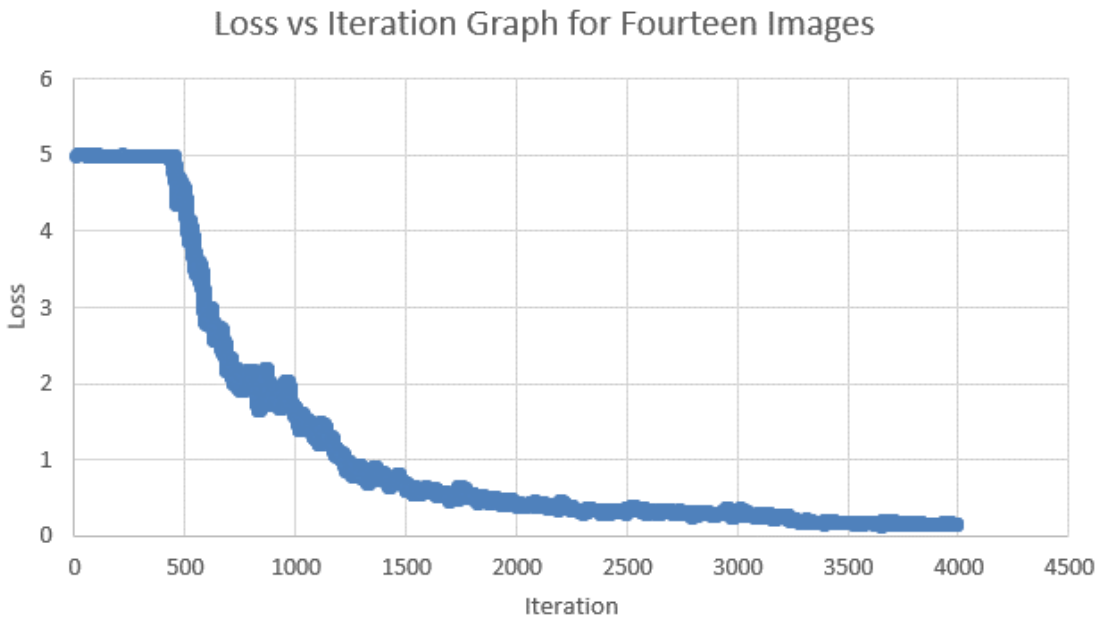
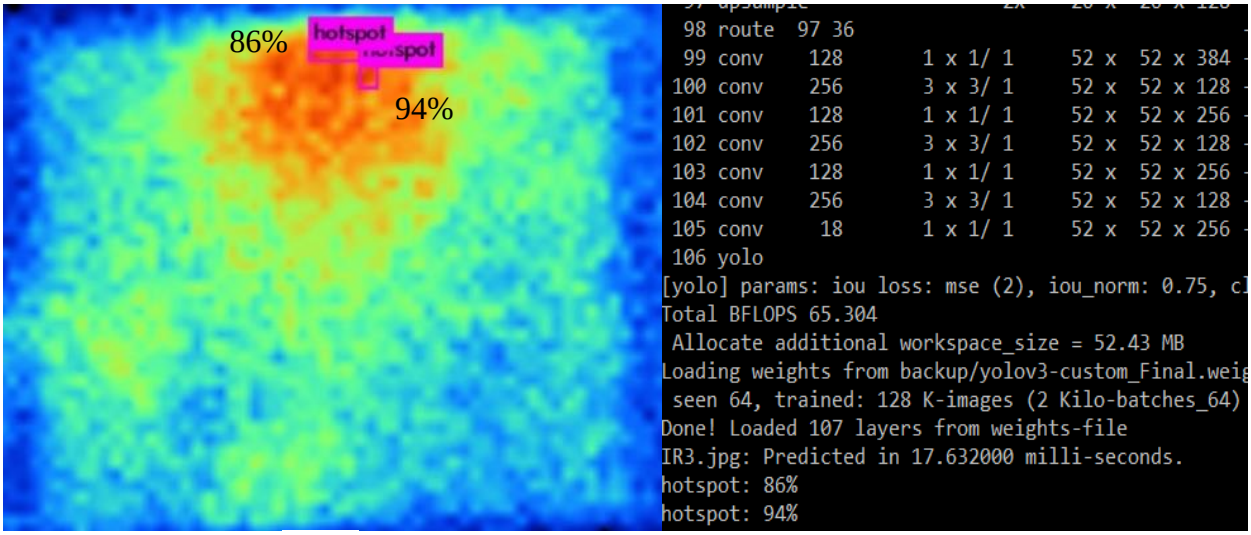


Fig 6. 10: Loss vs iteration graph of fourteen images

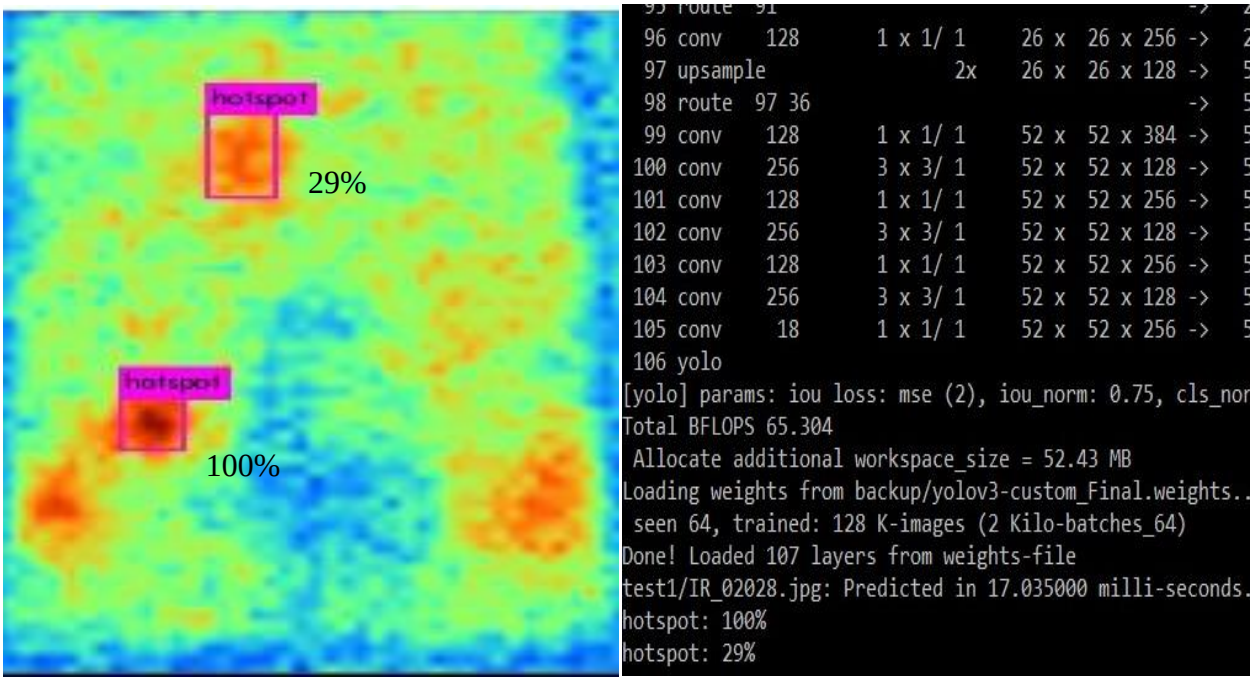
This time it took longer for the loss to start to decrease, since now there was a more diversified data set and training time was close to two hours. The more diversified the data set, the better will be the detection ability and the detector will be able to detect more hotspot regions. After the training finished, another weight file was generated which contain the learning knowledge of the detector gained from those images. The detector was tested using the new weight file and same test image as before and it is supposed to detected better, detect a few more hotspot regions. The predicted image is shown in Fig 6.11.



a b

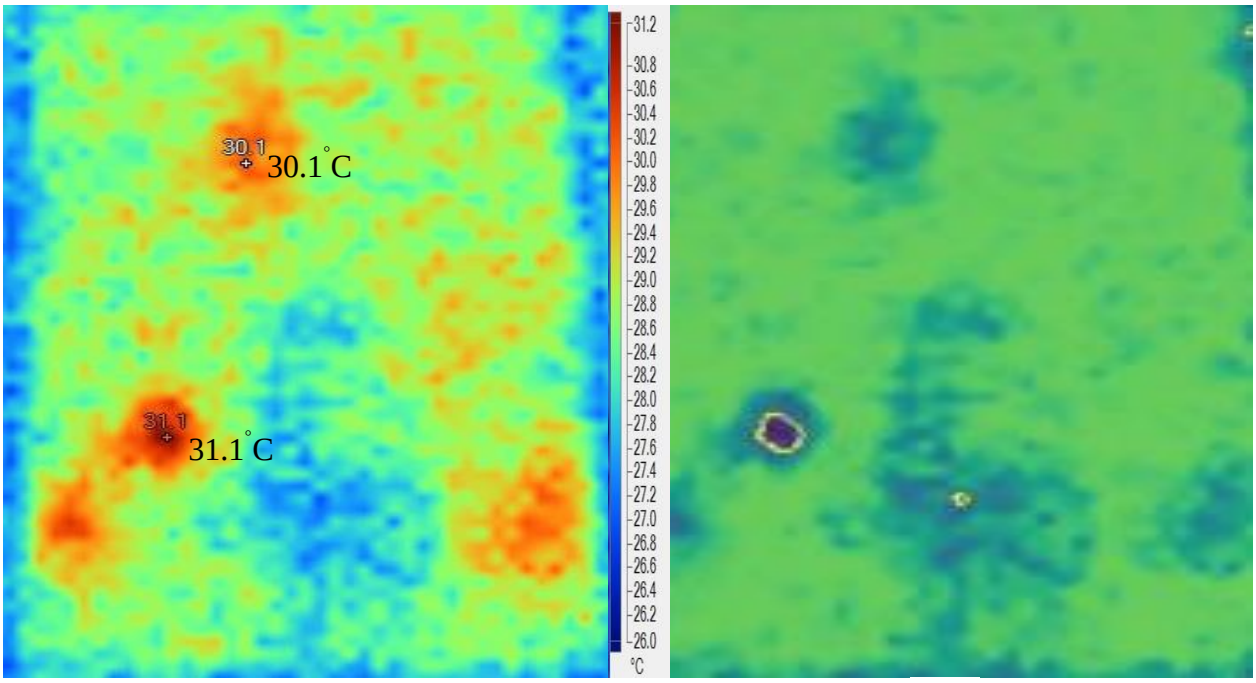
Fig 6. 11: a) Two hotspots detected on module- 13 and b) percentage of accuracy of the

By comparing Fig 6.2 with Fig 6.11 it is certain that the detector is getting better with more training data. The prediction shows that the number of predictions has increased as well the confidence of the detector has also increased from 56% to 94% for the same hotspot detected before. This time two hotspots are detected by increasing only four training images so it can be said that with the increase of training image detection will get better. Further, different thermal images were tested for better analysis. These test images are of different shapes, power and collected from previous study on thermal imaging. Fig 6.12 shows the output of test image- 1 and the detector has detected the highest temperature point of the module. This region is indeed a hotspot as shown in Fig 6.13. Similarly, it can be seen in Fig 6.14, Fig 6.16, Fig 6.18, Fig 6.20 detector has successfully detected the highest temperature point which are actually hotspots as shown in Fig 6.15, Fig 6.17, Fig 6.19, Fig 6.21.



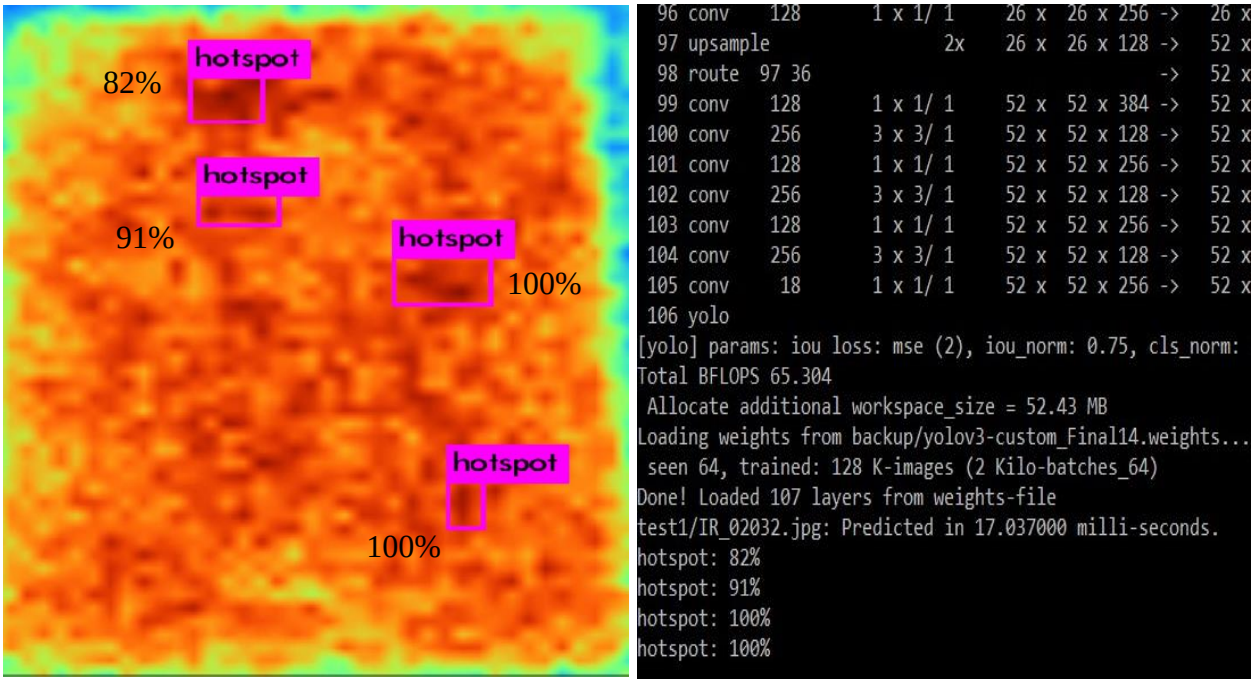
a b

Fig 6. 12: a) Two hotspots detected on Test image-1 and b) percentage of accuracy of the hotspot



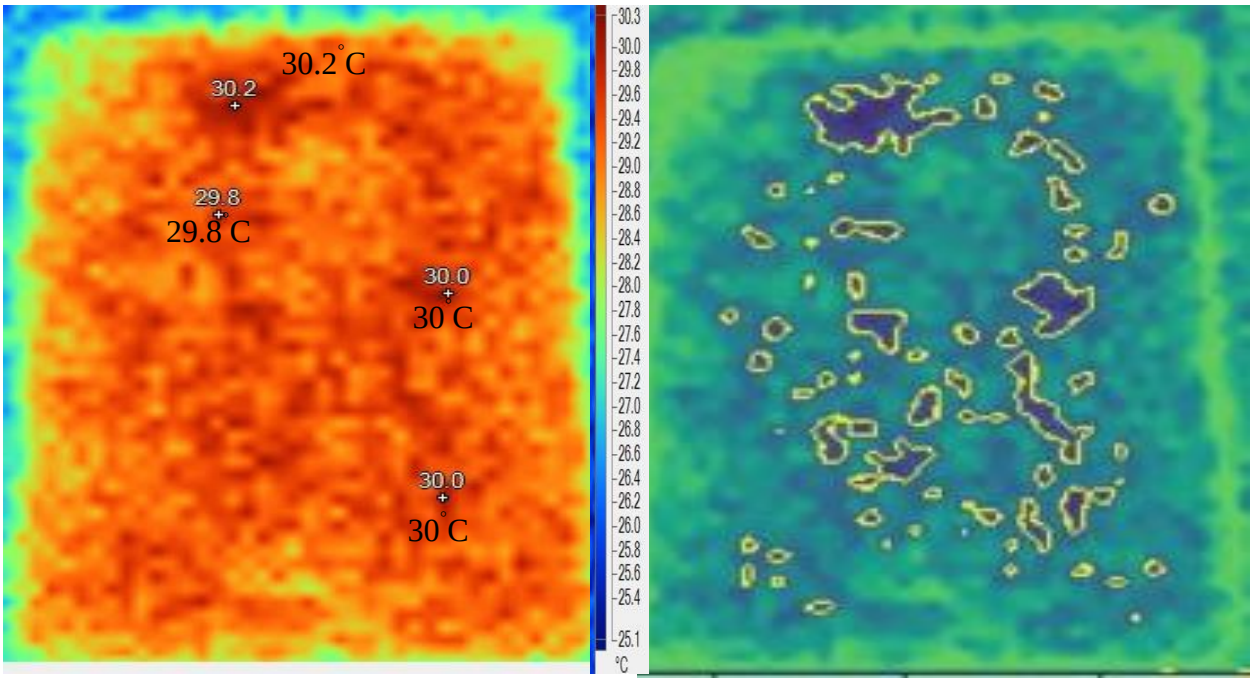
a b

Fig 6. 13: a) Temperature of the detected regions and b) actual hotspots of the Test image-1



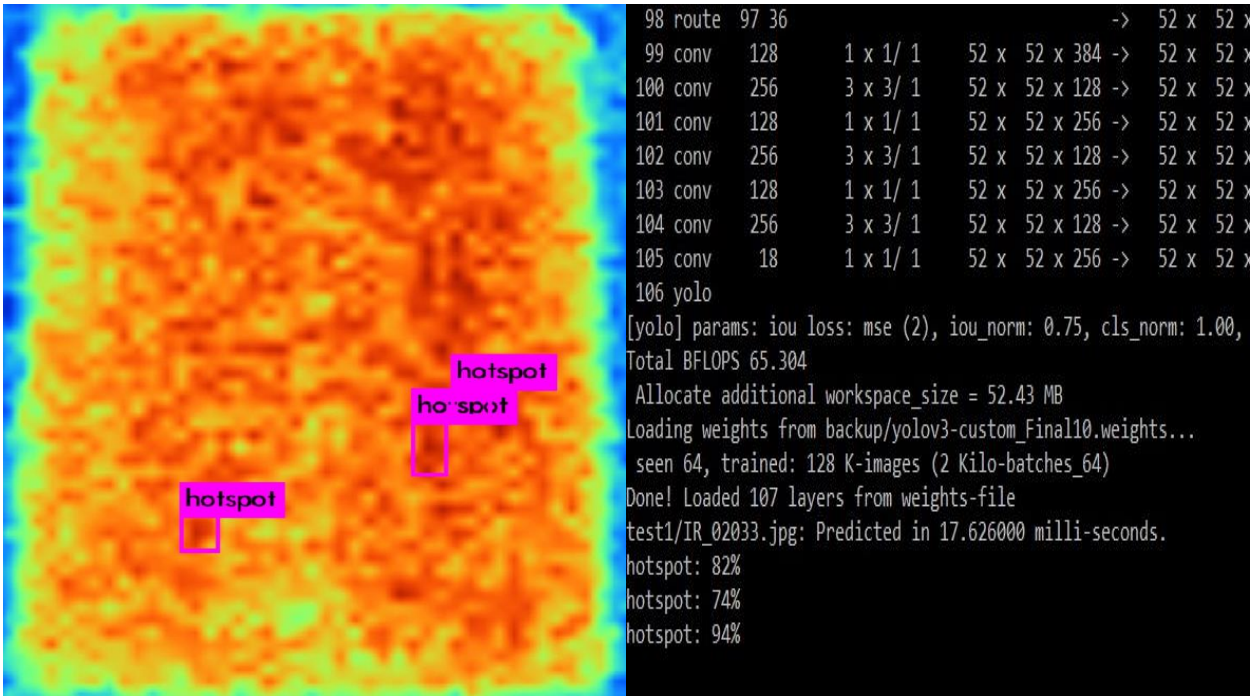
a b

Fig 6. 14: a) Four hotspots detected on test image 2 and b) percentage of accuracy of the hotspots



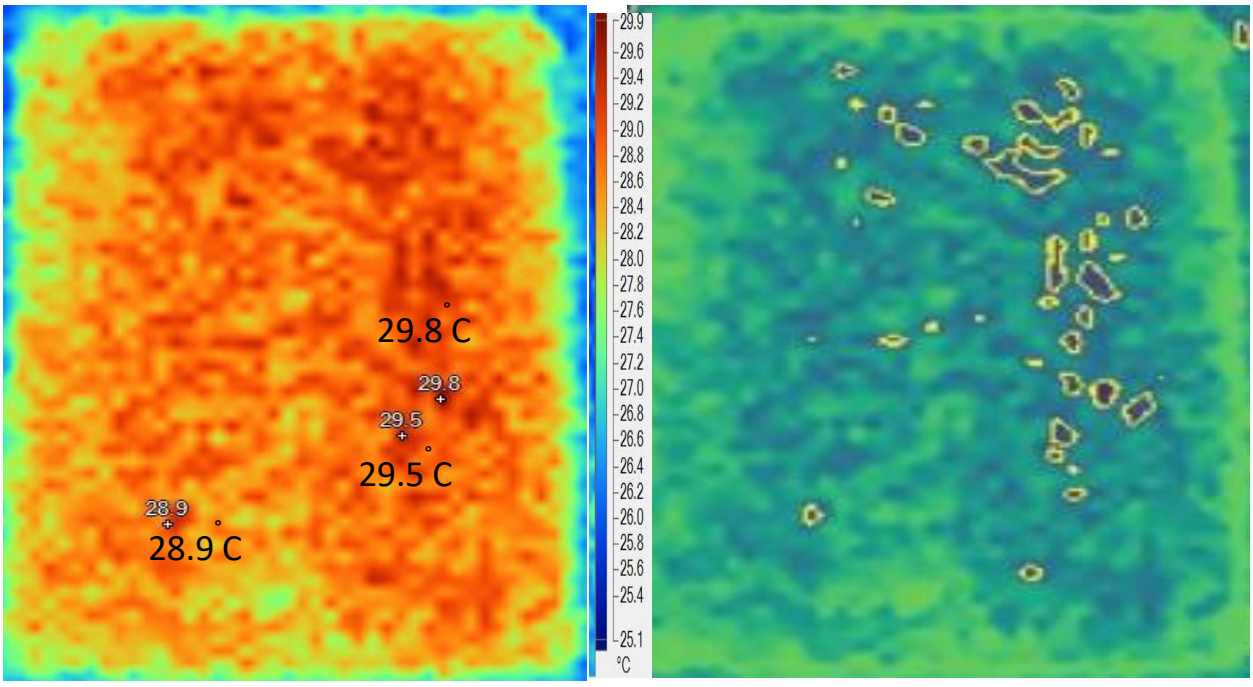
a b

Fig 6. 15: a) Temperature of the detected regions and b) actual hotspots of the test image 2



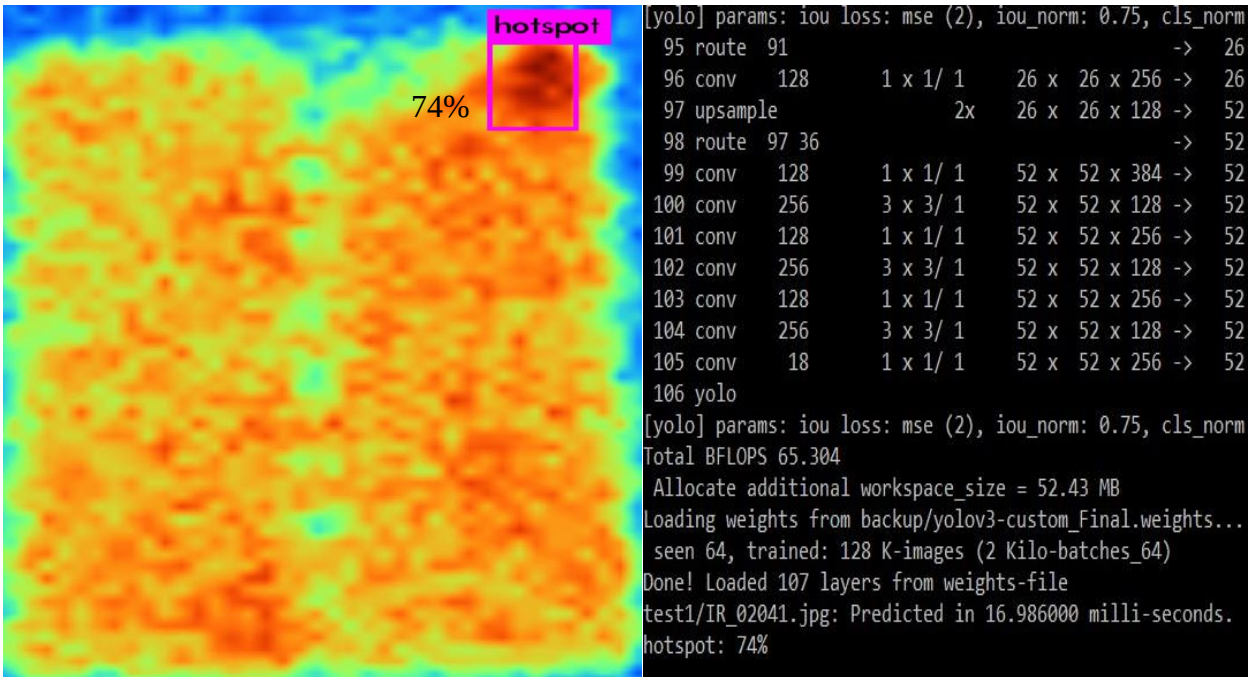
a b

Fig 6. 16: a) Two hotspots detected on test image 3 and b) percentage of accuracy of the hotspots



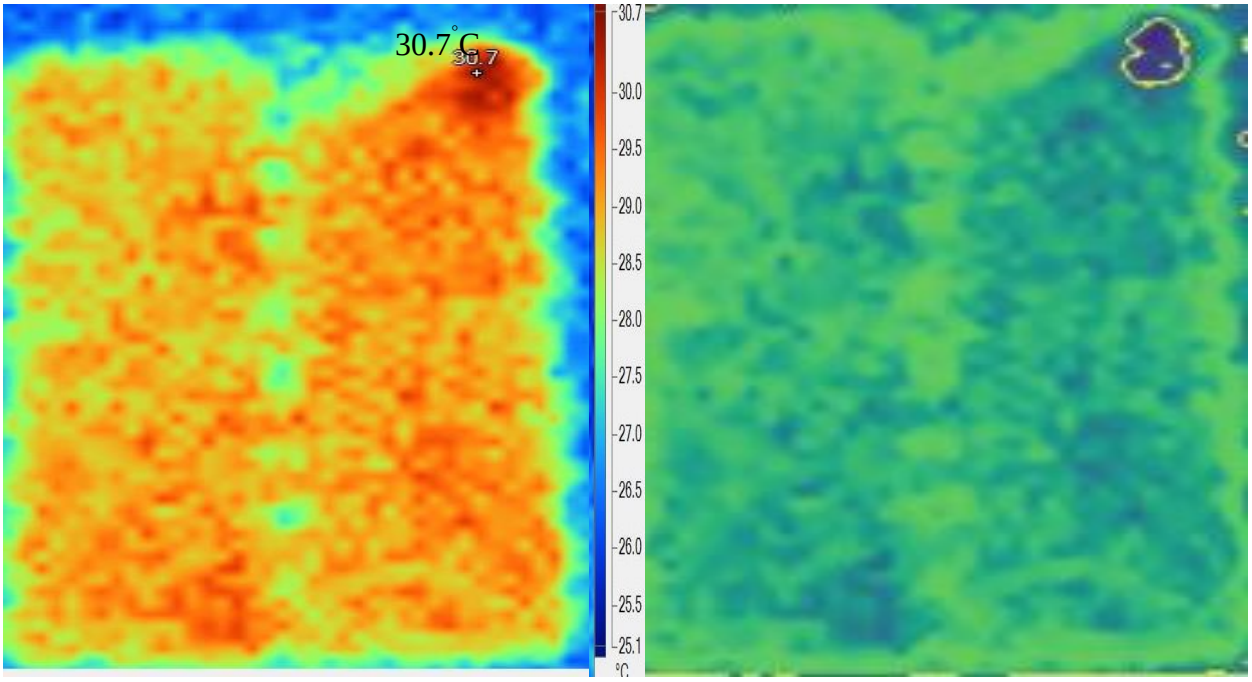
a b

Fig 6. 17: a) Temperature of the detected regions and b) actual hotspots of the test image 3



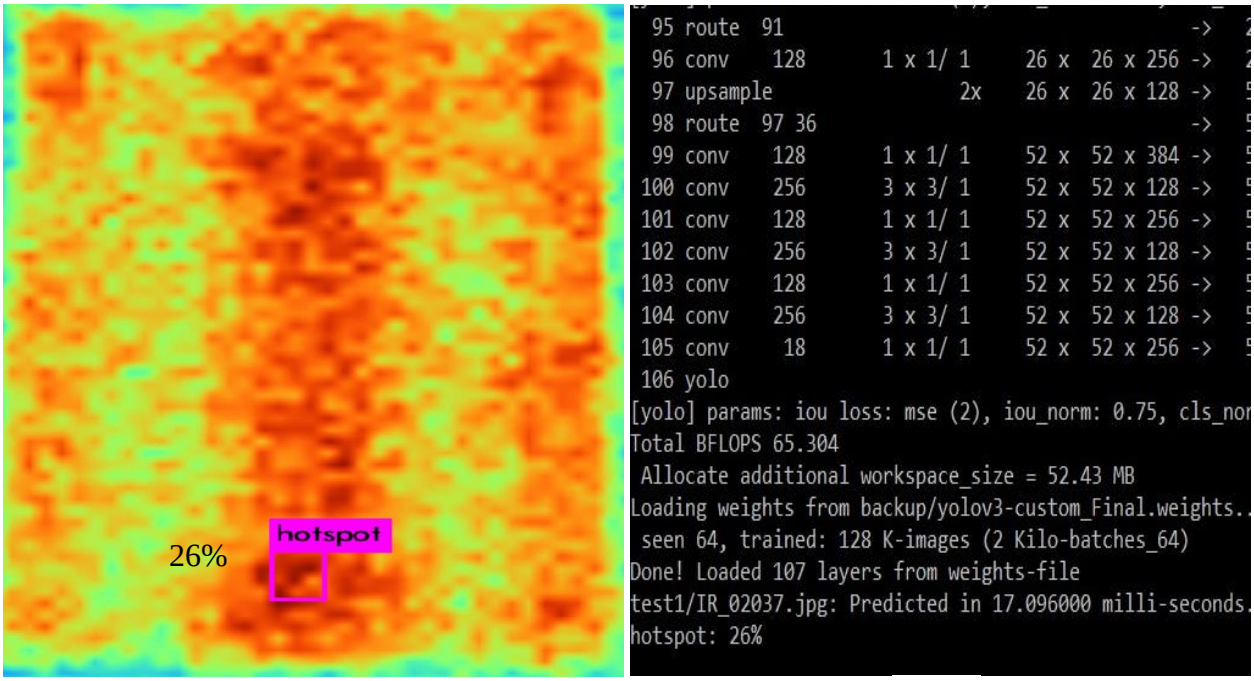
a b

Fig 6. 18: a) One hotspot detected on test image 4 and b) percentage of accuracy of the



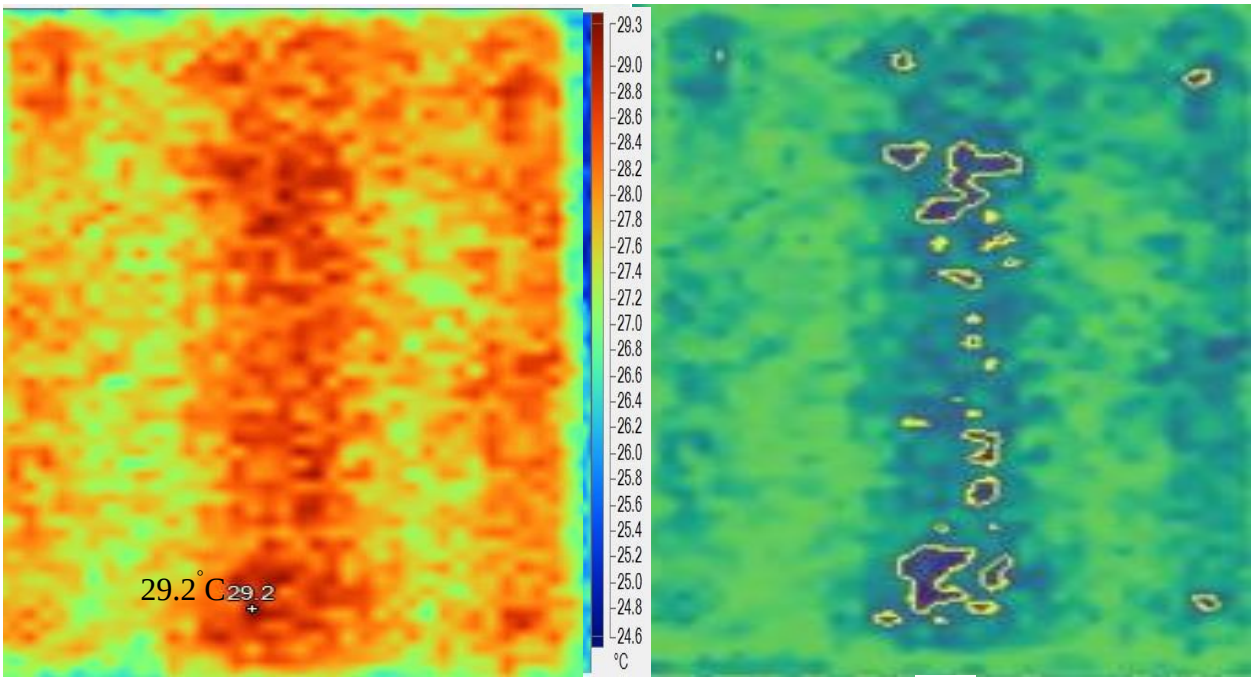
a b

Fig 6. 19:a) Temperature of the detected regions and b) actual hotspots of the test image 4



a b

Fig 6. 20: a) One hotspot detected on test image 5 and b) percentage of accuracy of the hotspot



a b

Fig 6. 21: a) Temperature of the detected regions and b) actual hotspots of the test image 5

In the following table the summary of the detector-2 is shown

Table 6. 2 Summarization of the detector-2 which is trained with Fourteen images

Test image	**Correct detection**	**Wrong detection**	**Confidence in the highest temperature region**
Test image-1	2	1	100%
Test image-2	4	0	82%
Test image-3	3	0	94%
Test image-4	1	0	74%
Test image-5	1	0	26%

As it is seen in “Test image-1,2” the confidence score on the highest temperature region increased. On the other hand, new hotspot regions are detected in detector-2 as seen in “Test image-3,4,5”.

6.4 Summary

From the above figure it is certain that the final detector which is trained with only fourteen images can detect hotspots. Nevertheless, the detector yet cannot detect all hotspots on a given module. As it is seen from two different training, the detector gets better if it gets more training data. So, if it is possible to feed more thermal images to the detector it will surely perform better.

Chapter 7

Conclusion

The aim of this work was to build a detector which could detect the hotspot of PV modules. Success was achieved after the algorithm detected the hotspot using the weight file (training data). At first with the help of IR camera taking of the IRT images of PV modules was done which helped validating the detected hotspot of PV modules & hence justified the proposed result of the classifier.

A part of this thesis work was to find relation using IV characteristics. Furthermore, based on IR images of modules it was found that temperature increase of the modules or the creation of hotspot is due to the impact of shunt & series resistance & found out that defected area & the average temperature difference doesn't have any interconnection.

The crucial part of this thesis work was to establish the detector. For setting up the detector few steps were followed. At the beginning dataset (IR images of PV modules) were processed manually for the YOLO & YOLO was trained, second part was the training of the dataset & final part was the testing of the dataset by the detector. At first IRT had been used in this work for taking the images of PV modules which was later used as the source of dataset & the confirmation for the detector to detect the right regions of hotspot. To get the result the detector was trained twice with datasets with 10 & 14 images respectively.

A single test image was tested with the 2-detector having 2 different weight file & as the training data increased the confidence of the detector increased. The final detector with highest datasets (14 images) showed highest improvement in terms of detecting accurate & a greater number of hotspots.

The result found after the training with 10 images showed that, the detector could actually detect the hotspot region with accuracy & when compared with actual image of the hotspot it could be seen that the detector was able to detect the hotspots meticulously, as it couldn't detect all the hotspots for having less dataset. But after training the detector with 14 images having new weight file containing all the learning knowledge from training, the detection quality improved. The confidence of the detector increased rapidly for the same hotspot region

when compared with prior detectors. Moreover, highest temperature point could be detected which when seen from the actual image showed it to be a hotspot.

Further, the time takes to detected a hotspot is around 17.5 millisecond which is significantly less. With this less detection time, with small modification real-time detection is possible. Thermal video of PV module can be feed into the detector and live detection is possible.

To conclude it can be said that the more diversified data the detector was feed into, the precision level of detection as well as the number of hotspot detection increased linearly & with more data the detector will improve the detection quality as well. So, keeping all these in mind while working with the PV modules in future & improving the detection ability of the detector this thesis work would be of great help.

References

1. A. Triki-Lahiani, A. Bennani-Ben Abdelghani and I. Slama-Belkhodja, "Fault detection and monitoring systems for photovoltaic installations: A review", *Renewable and Sustainable Energy Reviews*, vol. 82, pp. 2680-2692, 2018.
2. S. Firth, K. Lomas and S. Rees, "A simple model of PV system performance and its use in fault detection", *Solar Energy*, vol. 84, no. 4, pp. 624-635, 2010.
3. M. Alam, F. Khan, J. Johnson and J. Flicker, "PV faults: Overview, modeling, prevention and detection techniques", *2013 IEEE 14th Workshop on Control and Modeling for Power Electronics (COMPEL)*, pp. 1-7, 2013.
4. T. Sakagami and S. Kubo, "Applications of pulse heating thermography and lock-in thermography to quantitative nondestructive evaluations", *Infrared Physics & Technology*, vol. 43, no. 3-5, pp. 211-218, 2002.
5. V. Kurukuru, A. Haque, M. Khan and A. Tripathy, "Fault classification for Photovoltaic Modules Using Thermography and Machine Learning Techniques", *2019 International Conference on Computer and Information Sciences (ICCIS)*, pp. 1-6, 2019.
6. A. Oliveira, M. Aghaei, U. MAdukanya, R. *Rüther,* **"**FAULT INSPECTION BY AERIAL INFRARED THERMOGRAPHY IN APV PLANT AFTER A METEOROLOGICAL TSUNAMI **"**, pp. 17-25, 2019.
7. A. Oliveira, M. Aghaei and R. Ruther, "Automatic Fault Detection of Photovoltaic Arrays by Convolutional Neural Networks During Aerial Infrared Thermography", in *European PV Solar Energy Conference and Exhibition*, Marseille, France, 2019.
8. Y. He, B. Du and S. Huang, "Noncontact Electromagnetic Induction Excited Infrared Thermography for Photovoltaic Cells and Modules Inspection", *IEEE Transactions on Industrial Informatics*, vol. 14, no. 12, pp. 5585-5593, 2018.
9. G. Cipriani et al., "Application of Thermographic Techniques for the Detection of Failures on Photovoltaic Modules", *2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*, 2019.

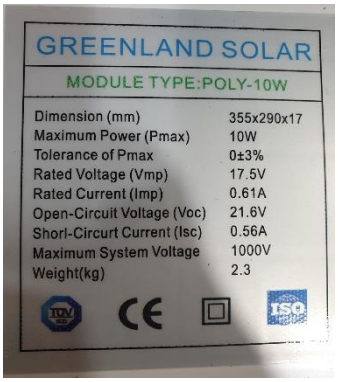
10. K. Niazi, W. Akhtar, H. Khan, S. Sohaib and A. Nasir, "Binary Classification of Defective Solar PV Modules Using Thermography", *2018 IEEE 7th World Conference on Photovoltaic Energy Conversion (WCPEC) (A Joint Conference of 45th IEEE PVSC, 28th PVSEC & 34th EU PVSEC)*, 2018.
11. J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779-788, 2016.
12. J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger", *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6517-6525, 2017.
13. Schuss et al., "Detecting Defects in Photovoltaic Cells and Panels and Evaluating the Impact on Output Performances," in IEEE Transactions on Instrumentation and Measurement, vol. 65, no. 5, pp. 1108-1119, May 2016.
14. G. Gomez, "Heat transfer in a photovoltaic panel", Lund University, Lund, Sweden, 2009.
15. M. Sabbaghpur Arani and M. Hejazi, "The Comprehensive Study of Electrical Faults in PV Arrays", *Journal of Electrical and Computer Engineering*, vol. 2016, pp. 1-10, 2016
16. A. Salazar and E. Macabebe, "Hotspots Detection in Photovoltaic Modules Using Infrared Thermography", *MATEC Web of Conferences*, vol. 70, pp. 1-5, 2016.
17. S. Djordjevic, D. Parlevliet and P. Jennings, "Detectable faults on recently installed solar modules in Western Australia", *Renewable Energy*, vol. 67, pp. 215-221, 2014.
18. J. Tsanakas and P. Botsaris, "Passive and Active Thermographic Assessment as a Tool for Condition-Based Performance Monitoring of Photovoltaic Modules", *Journal of Solar Energy Engineering*, vol. 133, no. 2, 2011.
19. C. Buerhop, H. Scheuerpflug, R. Weißmann, “The Role of Infrared Emissivity of Glass on IR-Imaging of PV-Plants”, Proc. 26th EUPVSEC (WIP, Hamburg, Germany), pp. 3413 – 3416, 2011.
20. Tensorflow.2020. *Introduction To Tensor*[online] Available at: https://www.tensorflow.org/guide/tensor
21. J. -H. Luo and J. Wu, "Neural Network Pruning With Residual-Connections and Limited-Data," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, pp. 3-11,26-29, 2020.

22. MissingLink.ai. 2020. *Fully Connected Layers In Convolutional Neural Networks: The Complete Guide- Missinglink.Ai.* [online] https://missinglink.ai/guides/convolutional-neural-networks/fully-connected-layers-convolutional-neural-networks-complete-guide/
23. S. Saha, "A Comprehensive Guide to Convolutional Neural Networks—the ELI5 way", *Medium*, 2018. [Online]. Available: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.
24. Dlology.com. 2017. *Gentle Guide On How YOLO Object Localization Works With Keras (Part 2) | Dlology*. [online] Available at: https://www.dlology.com/blog/gentle-guide-on-how-yolo-object-localization-works-with-keras-part-2/
25. Maj, M., 2020. *Object Detection And Image Classification With YOLO - Kdnuggets*. [online] KDnuggets. Available at: https://www.kdnuggets.com/2018/09/object-detection-image-classification-yolo.html
26. S. Dotenco et al., "Automatic detection and analysis of photovoltaic modules in aerial infrared imagery", *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016.
27. T. Sakagami and S. Kubo, "Applications of pulse heating thermography and lock-in thermography to quantitative nondestructive evaluations", *Infrared Physics & Technology*, vol. 43, no. 3-5, pp. 211-218, 2002.
28. F. Freire, S. Melcher, C. Hochgraf and S. Kurinec, "Degradation analysis of an operating PV module on a Farm Sanctuary", *Journal of Renewable and Sustainable Energy*, vol. 10, no. 1, 2018.
29. J. Tsanakas, L. Ha and C. Buerhop, "Faults and infrared thermographic diagnosis in operating c-Si photovoltaic modules: A review of research and future challenges", *Renewable and Sustainable Energy Reviews*, vol. 62, pp. 695-709, 2016.
30. M. García, L. Marroyo, E. Lorenzo, J. Marcos and M. Pérez, "Observed degradation in photovoltaic plants affected by hot-spots", *Progress in Photovoltaics: Research and Applications*, vol. 22, no. 12, 2013.
31. R. Huang, J. Pedoeem and C. Chen, "YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers", *2018 IEEE International Conference on Big Data (Big Data)*, 2018.

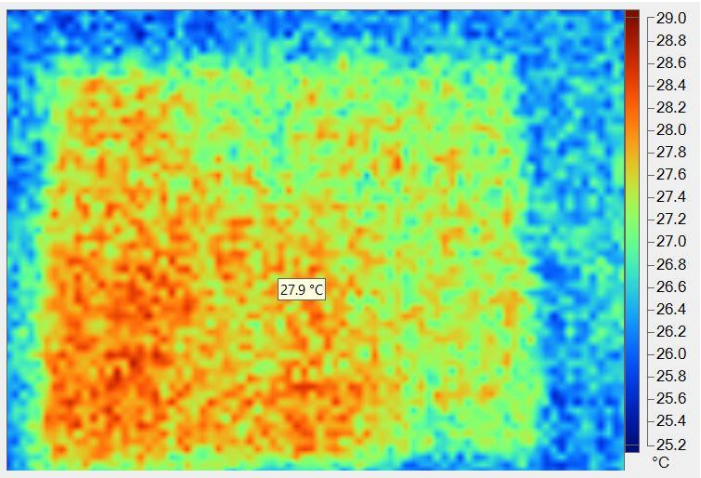
32. J. Barbedo, "Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification", *Computers and Electronics in Agriculture*, vol. 153, pp. 46-53, 2018.
33. G. shaha, "Real-Time Object Detection with Yolo", *International Journal of Engineering And Advanced Technology (IJEAT)*, vol. 8, no. 3, 2019.
34. K. Gurney, *An Introduction to Neural Networks*. London: University College London Press, 2001.
35. D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 2155–2162. IEEE, 2014.

Appendix

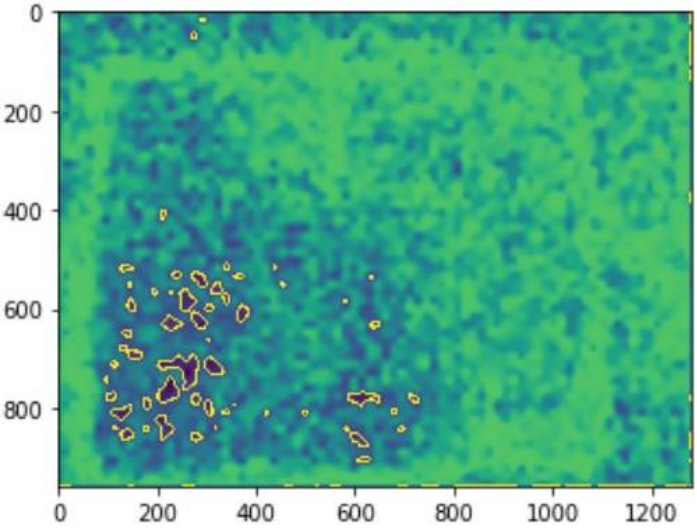
Specification of PV modules



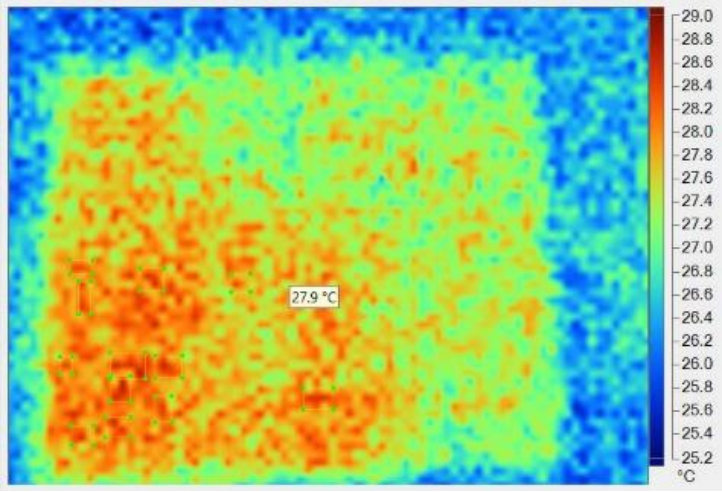
Whole Experiment in a brief step



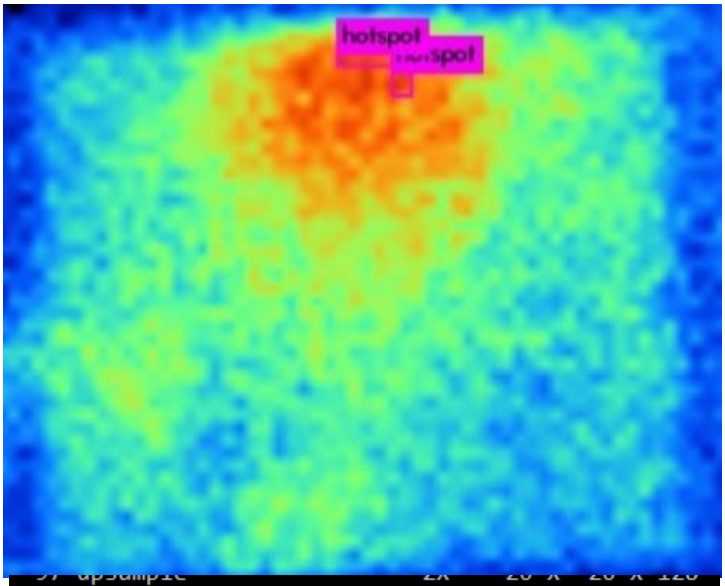
1. Thermal image of a PV modules



2. Processed thermal image for contouring the hotspots using python



3. Labelling the thermal image for training by comparing with the processed image



4. Output of YOLO detector after training with fourteen images.

```
  98 route  97 36
  99 conv    128       1 x 1/ 1     52 x  52 x 384
 100 conv    256       3 x 3/ 1     52 x  52 x 128
 101 conv    128       1 x 1/ 1     52 x  52 x 256
 102 conv    256       3 x 3/ 1     52 x  52 x 128
 103 conv    128       1 x 1/ 1     52 x  52 x 256
 104 conv    256       3 x 3/ 1     52 x  52 x 128
 105 conv     18       1 x 1/ 1     52 x  52 x 256
 106 yolo
[yolo] params: iou loss: mse (2), iou_norm: 0.75, cl
Total BFLOPS 65.304
 Allocate additional workspace_size = 52.43 MB
Loading weights from backup/yolov3-custom_Final.weig
 seen 64, trained: 128 K-images (2 Kilo-batches_64)
Done! Loaded 107 layers from weights-file
IR3.jpg: Predicted in 17.632000 milli-seconds.
hotspot: 86%
hotspot: 94%
```

5. Confidence of the detected hotspots on a testing image