

**DESIGN AND ANALYSIS OF PHOTOVOLTAIC ARRAY
TEMPERATURE ATTRIBUTES USING THERMAL IMAGING
SENSOR DEVICE FOR AI-BASED DEFECT DETECTION
SYSTEM**



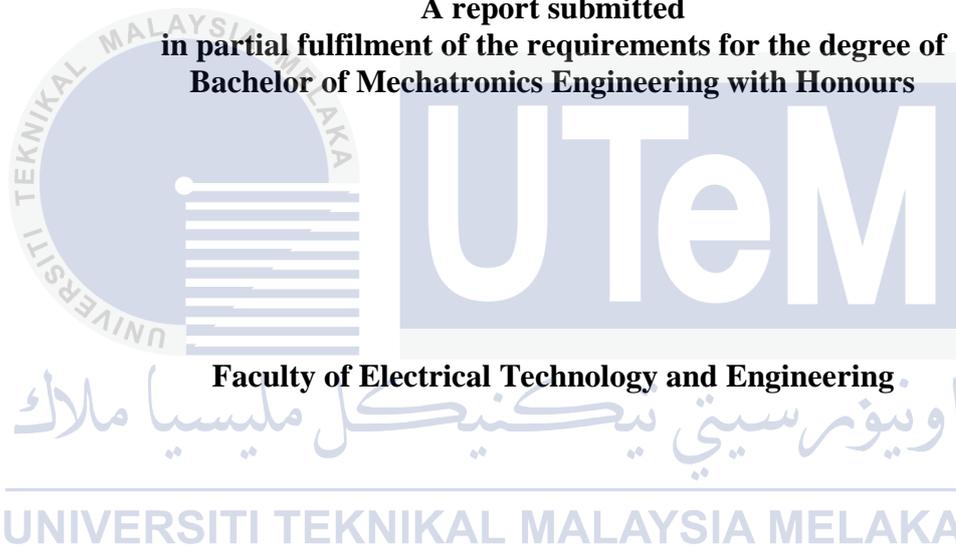
**BACHELOR OF MECHATRONICS ENGINEERING WITH
HONOURS
UNIVERSITI TEKNIKAL MALAYSIA MELAKA**

2024

**DESIGN AND ANALYSIS OF PHOTOVOLTAIC ARRAY TEMPERATURE
ATTRIBUTES USING THERMAL IMAGING SENSOR DEVICE FOR AI-BASED
DEFECT DETECTION SYSTEM**

LUQMAN AL HAKIM BIN KAMARUZAMAN

**A report submitted
in partial fulfilment of the requirements for the degree of
Bachelor of Mechatronics Engineering with Honours**



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

DECLARATION

I declare that this thesis entitled "DESIGN AND ANALYSIS OF PHOTOVOLTAIC ARRAY TEMPERATURE ATTRIBUTES USING THERMAL IMAGING SENSOR DEVICE FOR AI-BASED DEFECT DETECTION SYSTEM" is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in the candidature of any other degree.

Signature

:

Name

:

LUQMAN AL HAKIM BIN KAMARUZAMAN

Date

:

14 JUNE 2024



اوتيمر سیتی تیکنیکل ملیسیا ملاک

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

APPROVAL

I hereby declare that I have checked this report entitled “DESIGN AND ANALYSIS OF PHOTOVOLTAIC ARRAY TEMPERATURE ATTRIBUTES USING THERMAL IMAGING SENSOR DEVICE FOR AI-BASED DEFECT DETECTION SYSTEM”, and in my opinion, this thesis fulfils the partial requirement to be awarded the degree of Bachelor of Mechatronics Engineering with Honours

Signature :

Supervisor Name : PROFESOR MADYA DR. MUHAMMAD HERMAN BIN
JAMALUDDIN

Date :

20 JUNE 2024

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DEDICATIONS

To my beloved mother and father



ACKNOWLEDGEMENTS

In preparing this report, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thought. In particular, I wish to express my sincere appreciation to my main project supervisor, Professor Madya Dr. Muhammad Herman Bin Jamaluddin, for encouragement, guidance critics and friendship. Without his continued support and interest, this project would not have been same as presented here.



ABSTRACT

The need for sustainable and renewable energy alternatives is driven by the depletion of fossil fuels, which supply over 80% of the world's energy. Tidal, wind, geothermal, and solar energy are examples of clean energy, often known as renewable energy, because they can all be recycled naturally. As a renewable energy source, solar energy (including concentrating solar power (CSP) and solar photovoltaic (PV) power) contributes 3.6% of global electricity output. Nonetheless, it has cemented its position among other renewable energy technologies, accounting for more than 31% of total installed renewable energy capacity in 2022. Nonetheless, various faults, such as hotspots have an impact on the effectiveness and performance of solar panels. Underperformance of solar projects is becoming an increasing concern for solar energy system owners. The purpose of this study is to design a system that can easily detect defects on photovoltaic arrays of varying sizes and environmental conditions by developing an AI-based defect detection system using thermal imaging sensors to collect real-time temperature data from photovoltaic arrays and implement efficient algorithms for accurate defect identification, making it accessible and practical for the solar energy industry. This study is also to analyze the effectiveness and consistency of the AI-based defect detection system which uses YOLO v8. At first, The datasets are acquired from Google Images. Then, the datasets are acquired by flying a drone autonomously and capture thermal images of solar panels at a solar farm. The datasets are annotated using Roboflow. The AI model is trained and tested the AI model at 25, 50, 75 and 100 epochs. The effects of the number of epochs and the size of the datasets on the performance of the AI model was also analyzed. These findings are important in selecting the optimum object detection model.

ABSTRAK

Keperluan untuk alternatif tenaga yang mampan dan boleh diperbaharui didorong oleh kehabisan bahan api fosil, yang membekalkan lebih 80% tenaga dunia. Tenaga pasang surut, angin, geoterma dan suria adalah contoh tenaga bersih, selalunya dikenali sebagai tenaga boleh diperbaharui, kerana semuanya boleh dikitar semula secara semula jadi. Sebagai sumber tenaga boleh diperbaharui, tenaga suria (termasuk kuasa suria pemusatan (CSP) dan kuasa fotovoltaik suria (PV)) menyumbang 3.6% daripada keluaran elektrik global. Namun begitu, ia telah mengukuhkan kedudukannya di kalangan teknologi tenaga boleh diperbaharui yang lain, menyumbang lebih daripada 31% daripada jumlah kapasiti tenaga boleh diperbaharui terpasang pada tahun 2022. Namun begitu, pelbagai kerosakan, seperti titik panas mempunyai kesan ke atas keberkesanan dan prestasi panel solar. Prestasi projek suria yang kurang baik semakin menjadi kebimbangan bagi pemilik sistem tenaga suria. Tujuan kajian ini adalah untuk mereka bentuk sistem yang boleh mengesan kecacatan pada tatasusunan fotovoltaik dengan pelbagai saiz dan keadaan persekitaran dengan membangunkan sistem pengesanan kecacatan berasaskan AI menggunakan haba. penerima pengimejan untuk mengumpul data suhu masa nyata daripada tatasusunan fotovoltaik dan melaksanakan algoritma yang cekap untuk pengecaman kecacatan yang tepat, menjadikannya mudah diakses dan praktikal untuk industri tenaga suria. Kajian ini juga adalah untuk menganalisis keberkesanan dan ketekalan sistem pengesanan kecacatan berasaskan AI yang menggunakan YOLO v8. Pada mulanya, set data diperoleh daripada Google Images. Kemudian, set data diperoleh dengan menerbangkan dron secara autonomi dan menangkap imej haba panel solar di ladang solar. Set data dianotasi menggunakan Roboflow. Model AI dilatih dan menguji model AI pada 25, 50, 75 dan 100 zaman. Kesan bilangan zaman dan saiz set data pada prestasi model AI juga dianalisis. Penemuan ini penting dalam memilih model pengesanan objek yang optimum.

TABLE OF CONTENTS

	PAGE
DECLARATION	
APPROVAL	
DEDICATIONS	
ACKNOWLEDGEMENTS	2
ABSTRACT	3
ABSTRAK	4
TABLE OF CONTENTS	5
LIST OF TABLES	7
LIST OF FIGURES	9
LIST OF SYMBOLS AND ABBREVIATIONS	14
CHAPTER 1 INTRODUCTION	15
1.1 Background	15
1.2 Problem Statement	16
1.3 Motivation	17
1.4 Objective	19
1.5 Scope	19
1.6 Expected Results	20
CHAPTER 2 LITERATURE REVIEW	21
2.1 Renewable Energy	21
2.2 Solar Energy	23
2.3 Solar Photovoltaic (PV) Power	25
2.4 Defects on Photovoltaic Panels	33
2.4.1 Snail Trails	34
2.4.2 Hotspots	34
2.4.3 Shading and Shadowing	36
2.4.4 Microcracks	36
2.4.5 Delamination	37
2.4.6 Potential Induced Degradation (PID)	38
2.4.7 Diode Failure	39
2.5 Defect Detection	40
2.5.1 Previous Work on Defect Detection	41
2.6 Artificial Intelligence	47
2.7 You Only Look Once (YOLO)	49
2.7.1 YOLO v5 vs YOLO v8	51
2.8 Thermal Imaging Sensor.	54
2.8.1 Principle of Thermal Sensing	55

2.8.2	Latest Developments in Thermal Cameras	57
2.9	Drone	59
2.10	Summary	61
CHAPTER 3 METHODOLOGY		63
3.1	Introduction	63
3.2	Flowchart	63
3.3	Acquiring Dataset	64
3.4	Annotating Dataset	66
3.5	Structure of Dataset	69
3.6	Training and Testing	71
	3.6.1 Google Collab	71
	3.6.2 Offline	73
3.7	Summary	76
CHAPTER 4 RESULTS AND DISCUSSIONS		78
4.1	Custom Training Dataset	78
	4.1.1 Google Collab	78
	4.1.2 Offline	80
4.2	Confusion Matrix and Graph	83
	4.2.1 Google Collab	85
	4.2.2 Offline	92
4.3	Validation and Inference Results	99
	4.3.1 Google Collab	100
	4.3.2 Offline	103
4.4	Comparison Between Epochs and Dataset	105
	4.4.1 Comparison Between Epochs	106
	4.4.2 Comparison Between Size of Datasets	108
4.5	Graphical User Interface (GUI)	111
CHAPTER 5 CONCLUSION AND RECOMMENDATIONS		118
5.1	Conclusion	118
5.2	Future Work	120
REFERENCES		122

LIST OF TABLES

Table 1.1 Global installed solar capacity from 2013 to 2022	18
Table 1.2 Top fourteen solar energy installers in 2022	18
Table 2.1 Lists of solar energy technology classification	24
Table 2.2 Global installed solar PV capacity from 2013 to 2022.	27
Table 2.3 Top fourteen solar PV energy installers in 2022.	28
Table 2.4 Comparison of previous work.	47
Table 2.5 The evolution of YOLO object detection versions.	51
Table 2.6 Comparison of experimental result.	53
Table 2.7 Object detection performance comparison between YOLO v8 and YOLO v5.	53
Table 2.8 Subbands in the infrared spectrum.	54
Table 2.9 Popular thermal cameras and their specifications.	58
Table 2.10 Drones that are available in the market.	60
Table 2.11 Drones that are available in the market.	60
Table 3.1 Structure of dataset for both FYP	70
Table 4.1 Method and time taken for training and testing	83
Table 4.2 Confusion matrix	85
Table 4.3 Results for all epochs (Google Collab)	91
Table 4.4 Number of images detected for all epochs (Google Collab)	91
Table 4.5 Performance metrics for all epochs (Google Collab)	91
Table 4.6 Results for all epochs (offline)	98
Table 4.7 Number of images detected for all epochs (offline)	98
Table 4.8 Performance metrics for all epochs (offline)	98

Table 4.9 Confusion Matrix at 75 epochs

109

Table 4.10 Performance at 75 epochs

110



LIST OF FIGURES

Figure 1.1 The contribution of energy sources in both electricity generation and total installed power capacity by 2050	18
Figure 2.1 The growth of renewable energy	22
Figure 2.2 Yearly world solar photovoltaic estimated deployments from 2000–2050 [64]	24
Figure 2.3 The world solar PV installed capacity by 2050.	28
Figure 2.4 The global weighted-average total installed cost of solar PV projects since 2014, followed by 2050.	29
Figure 2.5 The total installed cost trends for solar PV projects in major markets since 2010.	30
Figure 2.6 The global levelized cost of electricity for solar PV projects since 2010, followed by 2050.	31
Figure 2.7 Regional weighted average levelized cost of electricity for solar PV projects in the major markets since 2010.	32
Figure 2.8 The global weighted average LCOE, capacity factor, and total installed costs for CSP projects between 2010 and 2022.	33
Figure 2.9 Snail trails defect on a photovoltaic panel.	34
Figure 2.10 Hotspots defect on a photovoltaic panel.	35
Figure 2.11 Hotspots defect on a photovoltaic panel under thermal imaging.	35
Figure 2.12 Hotspots defect on a photovoltaic panel under thermal imaging.	35
Figure 2.13 Shading and shadowing defect on photovoltaic panels.	36
Figure 2.14 Microcracks defect on photovoltaic panels.	37
Figure 2.15 Delamination defect on a photovoltaic panel.	38

Figure 2.16 Before and after PID defect on a photovoltaic panel.	39
Figure 2. 17 Diode failure defect on a photovoltaic panel .	40
Figure 2.18 EL imaging setup.	44
Figure 2.19 Examples on EL images taken form the ELPV dataset: a) mono c-Si cells (mc-Si), and b) poly c-Si cells (pc-Si).	44
Figure 2.20 Image classification result.	45
Figure 2.21 Accuracy results.	45
Figure 2.22 Loss results.	45
Figure 2.23 YOLO v5 network structure.	46
Figure 2.24 The relationship between Artificial Intelligence, Machine Learning, Deep Learning, and Explainable Artificial Intelligence.	49
Figure 2.25 Process of Machine Learning.	49
Figure 2.26 Working of the bolometer-based thermal sensor.	56
Figure 2.27 Components, selection criteria, and classification of thermal cameras.	59
Figure 3.1 Flowchart for model generation.	63
Figure 3.2 Drone Harmony route planning	65
Figure 3.3 Drone take off	65
Figure 3.4 Drone flying autonomously above the photovolltaic panel	66
Figure 3.5 Lux meter with 55700 lux	66
Figure 3.6 Roboflow tool.	67
Figure 3.7 Roboflow tool (Uploading images).	68
Figure 3.8 Roboflow tool (Annotating images).	68
Figure 3.9 Annotation of hotspots	69
Figure 3.10 Annotation of hotspots	69
Figure 3.11 Structure of dataset for FYP 2	71

Figure 3.12 Training and testing using Goolge Collab	72
Figure 3.13 Installing and importing Roboflow	72
Figure 3.14 Code to train the model using Google Collab	72
Figure 3.15 Command prompt to train offline	74
Figure 3.16 Steps to train and test using CPU	74
Figure 3.17 Python environment activated	75
Figure 3.18 Steps to convert CPU to GPU	75
Figure 3.19 GPU is available	76
Figure 3.20 Code to train the model using Offline	76
Figure 4.1 Training and testing process in Google Collab	78
Figure 4.2 Training and testing process in Google Collab	79
Figure 4.3 Training and testing process in Google Collab	79
Figure 4.4 Training and testing process in Google Collab	80
Figure 4.5 Training and testing process in Google Collab	80
Figure 4.6 Training and testing process via offline	81
Figure 4.7 Training and testing process via offline	81
Figure 4.8 Training and testing process via offline	82
Figure 4.9 Training and testing process via offline	82
Figure 4.10 Confusion matrix at 25 epochs (Google Collab)	87
Figure 4.11 Results graph at 25 epochs (Google Collab)	87
Figure 4.12 Confusion matrix at 50 epochs (Google Collab)	88
Figure 4.13 Results graph at 50 epochs (Google Collab)	88
Figure 4.14 Confusion matrix at 75 epochs (Google Collab)	89
Figure 4.15 Results graph at 75 epochs (Google Collab)	89
Figure 4.16 Confusion matrix at 100 epochs (Google Collab)	90

Figure 4.17 Results graph at 100 epochs (Google Collab)	90
Figure 4.18 Confusion matrix at 25 epochs (Offline)	94
Figure 4.19 Results graph at 25 epochs (Offline)	94
Figure 4.20 Confusion matrix at 50 epochs (Offline)	95
Figure 4.21 Results graph at 50 epochs (Offline)	95
Figure 4.22 Confusion matrix at 75 epochs (Offline)	96
Figure 4.23 Results graph at 75 epochs (Offline)	96
Figure 4.24 Confusion matrix at 100 epochs (Offline)	97
Figure 4.25 Results graph at 100 epochs (Offline)	97
Figure 4.26 Valdiation code	99
Figure 4.27 Inference code	100
Figure 4.28 Result at 25 epochs (Google Collab)	100
Figure 4.29 Result at 50 epochs (Google Collab)	101
Figure 4.30 Result at 75 epochs (Google Collab)	101
Figure 4.31 Result at 100 epochs (Google Collab)	102
Figure 4.32 Result at 25 epochs (Offline)	103
Figure 4.33 Result at 50 epochs (Offline)	104
Figure 4.34 Result at 75 epochs (Offline)	104
Figure 4.35 Result at 100 epochs (Offline)	105
Figure 4.36 Comparision between epoch with 339 datasets	107
Figure 4.37 Comparision between epoch with 3064 datasets	108
Figure 4.38 Results graph at 75 epochs with 339 datasets	110
Figure 4.39 Results graph at 75 epochs with 3064 datasets	110
Figure 4.40 GUI design using Make Real.Tldraw	113
Figure 4.41 GUI first stage	114

Figure 4.42 GUI second stage	114
Figure 4.43 GUI third stage	115
Figure 4.44 GUI fourth stage	115
Figure 4.45 GUI fifth stage	116
Figure 4.46 GUI final stage	116
Figure 4.47 Popup message	117
Figure 4.48 Popup message	117



LIST OF SYMBOLS AND ABBREVIATIONS

W	-	Energy flux
k	-	Boltzmann's constant
σ	-	Stefan-Boltzmann's constant
h	-	Planck's constant
T	-	Kelvin
ϵ	-	Emissivity
c	-	Speed of light in vacuum



CHAPTER 1

INTRODUCTION

1.1 Background

The depletion of fossil fuels, which account for more than 80% of global energy consumption, needs prompt research into sustainable and renewable energy alternatives. Clean energy, or renewable energy, is defined as energy that can be recycled in nature, such as tidal energy, wind energy, geothermal energy, and solar energy. In comparison to the traditional non-renewable energy that is dwindling, it is endless, and its regeneration is automatic, without human intervention, and will not cause undue environmental damage [1]. Renewable energy is gaining popularity as a future energy source around the world. Solar energy is a readily available, sustainable, and renewable energy source. Solar energy (includes concentrating solar power (CSP) and solar photovoltaic (PV) power) contributes 3.6% of worldwide electricity output as a renewable energy source. It has, nevertheless, solidified its position among other renewable energy technologies, contributing for over 31% of total installed renewable energy capacity in 2022. With a capacity of 1053 GW in 2022, solar energy is the second most deployed renewable energy technology, trailing only hydroelectric technology [2].

Solar energy, which originates from the sun in the form of solar irradiance, can be transformed directly to electricity using photovoltaic (PV) technology. PV technology employs semiconductor-based solar cells to capture solar radiation and convert it to electrical energy [3]. Solar energy has garnered a lot of attention in recent years as a feasible replacement for fossil fuels. It is a renewable and sustainable energy source. Nonetheless, various faults, such as cells, diodes, or multiple cells and multiple diodes, may have an impact on the effectiveness and performance of solar panels. These defects can considerably impair energy generation; hence it is vital to discover, categories and overcome them as soon as possible to avoid loss of money [4].

1.2 Problem Statement

Solar energy is a widely available, sustainable, and renewable source of energy. Solar energy, being a renewable resource, has the potential to replace the extensively used fossil fuel supply in the near future [2]. Solar energy has garnered a lot of attention in recent years as a feasible replacement for fossil fuels. It is a renewable and sustainable energy source. Nonetheless, various faults, such as cells, diodes, or multiple cells and multiple diodes, may have an impact on the effectiveness and performance of solar panels. These defects can considerably impair energy generation; hence it is vital to discover, categories and overcome them as soon as possible to avoid loss of money [4].

Underperformance of solar projects is becoming an increasing concern for solar energy system owners. Underperformance from anomalies roughly doubled from 2019 to 2022, from 1.61% to 3.13%, according to Raptor Maps data from analyzing 24.5 GW of large-scale solar systems in 2022. As systems age, solar panel underperformance due to equipment-related downtime and anomalies becomes more typical. Unfortunately, these challenges come at a high financial cost. The predicted annual income loss from the study sample's 24.5 GW was \$82 million, resulting in a \$2.5 billion loss for the whole solar energy business. According to the 2023 Solar Risk Assessment, there are various potential concerns that could impair solar system performance. Extreme weather risks, the increased use of solar systems in harsh weather areas, and the difficulty of estimating equipment-related performance are all significant considerations [5].

Another study by The National Renewable Energy Laboratory (NREL) provides a more realistic image of the deterioration of solar panels. In this study, which examined the rates of deterioration for nearly 2,000 solar systems in various regions around the globe, it was discovered that monocrystalline panels manufactured after 2000 only declined at a rate of 0.4%, which is less than half of the 1% rate specified in the warranties [6].

1.3 Motivation

Solar energy has received a lot of attention in recent years as a viable alternative to fossil fuels. It is a renewable and long-term energy source [4]. However, every item that is created will deteriorate over time. Solar panels are no exception. It will defect in the future. These flaws can significantly decrease energy generation; thus, it is critical to identify, categorize, and overcome them as soon as possible to avoid financial loss. As mentioned previously, according to Raptor Maps data from analyzing 24.5 GW of large-scale solar systems in 2022, underperformance from anomalies roughly doubled from 2019 to 2022, from 1.61% to 3.13% [5].

Tables 1.1 and 1.2 illustrate the global installed solar capacity during the last 10 years, as well as the contributions of the top fourteen countries. Table 2.1 indicates a massive 22% rise in solar energy installed capacity between 2021 and 2022. While the top three installers are China, the United States, and Japan, China's relative contribution accounts for approximately 37% of total solar installation in 2022. Figure 2.2 depicts the contribution of energy sources to total installed power capacity and electricity generation by 2050. As seen in Figure 2.2, renewables accounted for almost 30% of global installed capacity in 2016, accounting for nearly a quarter of global energy generation. Solar power (PV+CSP) contributed roughly 8% of renewable electricity output. As illustrated in Figure 2.2, solar PV technology is estimated to have the most installed capacity (8519 GW) by 2050, making it the second most dominant production source after wind power, and to generate nearly 25% of total electricity needs by 2050 [2]. For this paper, the researcher will be focusing on solar photovoltaic (PV) power as it is the most common type of solar energy.

The precision of the current solar array defect detection techniques is frequently lacking, which lowers energy output and increases maintenance expenses. Variations in temperature between arrays can be useful markers of faults, but current systems are unable to take advantage of this feature since real-time thermal data is not available. The goal of this project is to create a complete and affordable system that combines thermal imaging sensors with AI algorithms to provide reliable flaw identification and ongoing monitoring. In addition, it addresses issues with data quality, real-time analysis, and reducing false positives, all of which lead to the production of solar energy that is more dependable and efficient.

Table 1.1 Global installed solar capacity from 2013 to 2022

	Solar energy capacity (MW)									
	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
World	140 514	180 712	228 920	301 082	395 947	489 306	592 245	720 429	861 537	1 053 115
Africa	716	1 709	2 242	3 455	5 200	8 150	9 493	10 819	11 628	12 641
Asia	36 225	60 691	90 581	140 489	211 853	276 406	332 854	410 326	485 413	597 573
Europe	84 189	91 095	99 604	106 173	112 299	121 603	142 272	162 795	190 143	227 799
N. America	13 645	20 129	27 043	38 731	47 828	57 664	69 656	86 493	107 192	126 443
S. America	198	465	921	1 589	3 672	5 512	8 562	13 164	20 795	32 773
Oceania	4 610	5 358	6 079	6 860	7 576	8 881	13 293	18 357	23 342	27 400

Table 1.2 Top fourteen solar energy installers in 2022

N/s	Country	Installed capacity (GW)
1	China	393.0
2	USA	113.1
3	Japan	78.8
4	Germany	66.5
5	India	63.1
6	Australia	26.8
7	Italy	25.1
8	Brazil	24.1
9	Netherlands	22.6
10	Korea Rep	20.9
11	Spain	20.5
12	Viet Nam	18.5
13	France	17.4
14	UK	14.4

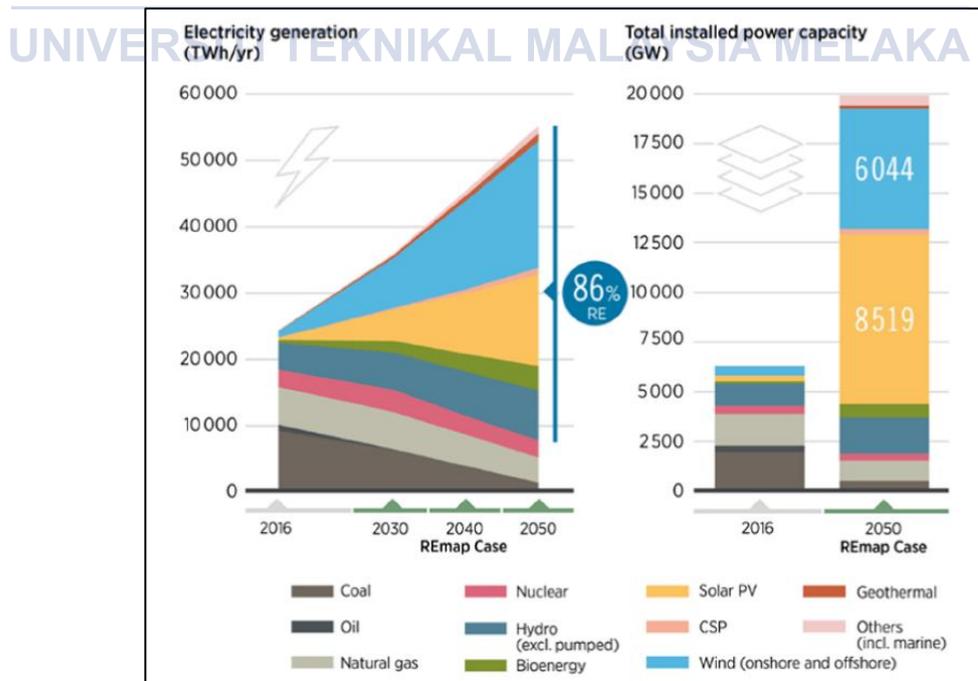


Figure 1.1 The contribution of energy sources in both electricity generation and total installed power capacity by 2050

1.4 Objective

Our objectives for this project are:

- i. To design a system that can easily detect defects on photovoltaic arrays of varying sizes and environmental conditions
- ii. To develop a model for AI-based defect detection system that refers to thermal imaging sensors
- iii. To analyze the effectiveness and consistency of the AI-based defect detection system which uses YOLO v8.

1.5 Scope

- i. The model is mainly for detecting defect on photovoltaic arrays of varying sizes and environmental conditions.
- ii. The model uses thermal imaging sensor to collect real-time temperature data from the photovoltaic arrays and will not be using any other types of sensors to collect data.
- iii. The model is an AI-based defect detection system that uses YOLO v8 which is the latest YOLO v Series. This model will not focus on other types of AI-based defect detection system such as TensorFlow.
- iv. The model will be using a drone to move and position the thermal imaging sensor. The drone is self-operated.
- v. The model will be solely focus on the development of the AI-based defect detection system and not the development of the drone and thermal imaging sensor.
- vi. The model will be solely detecting defects on photovoltaic arrays by its temperature attributes. This model will not detect cracks and other types of defects.
- vii. The model will be using a drone that is already available in the market and will not make any enhancement or improvement on the drone. There also will be no any type of development related to drone.
- viii. The model will be using a thermal imaging sensor that is already available in the market and will not make any enhancement or improvement on the thermal imaging sensor. There also will be no any type of development related to thermal imaging sensor.

- ix. The model will capture the thermal imaging images on a bright sunny day with a luminescent of 20000 to 60000 lux.

1.6 Expected Results

At the end of the project, it is expected that the Design and Analysis of Photovoltaic Array Temperature Attributes Using Thermal Imaging Sensor Device for AI-based Defect Detection System can:

- i. Detect defects on photovoltaic arrays of varying sizes and environmental conditions, making it accessible and practical for the solar energy industry.
- ii. Enhanced defect detection accuracy by using an AI-based defect detection system using thermal imaging sensors to collect real-time temperature data from photovoltaic arrays and implement efficient algorithms for accurate defect identification.
- iii. Achieved a effective and consistent result of the AI-based defect detection system which uses YOLO v8.

CHAPTER 2

LITERATURE REVIEW

2.1 Renewable Energy

As an alternative to conventional energy sources, the usage of renewable energy sources is growing quickly [7]. Renewable energy is energy generated from naturally replenished sources such as the sun and wind. Renewable energy can be used to generate power, heat and cool buildings, and move people [8]. Throughout human history, energy has been a vital subject that has almost completely surrounded existence. Although conventional energy resources such as coal, oil, and gas have been widely used to supply energy, oil reserves run out in 50 years, gas reserves last only 70 years, and coal reserves are totally depleted in 200 years. As a result, the traditional world is looking for alternative energy sources. Some countries raise awareness about renewable energy in order to kickstart the new eco-friendly energy usage. The primary goal of energy policy is to obtain energy in a dependable, continuous, clean, and cost-effective manner, while also expanding the sources [9].

The current global energy system is unsustainable, and transitioning to renewable energy could benefit both people and the environment [10]. Promoting renewable energy policy can improve equity, health, and employment while reducing greenhouse gas emissions [11]. But the debate about switching to only renewable energy can get divisive, and some scholars have argued that it is not conceivable from an institutional, technological, or economic standpoint [12], [13], [14]. Such arguments highlight concerns about how markets can include energy from renewable sources, potential environmental effects, and the capacity to deal with the frequently ephemeral character of these sources. On the other hand, it has been stated that it is technically and economically possible to switch to a system that uses only renewable energy [15], [16], [17]. It has been suggested that these kinds of systems will provide advantages including lower levelized energy costs and lower water usage. Studies analysing a transition to 100% renewable energy have examined the electrification of every industry, including transportation, heating, and desalination. It has been demonstrated that electrification from renewable sources allows for more flexibility, which

results in efficient systems. The technical and financial advantages of an electricity system based on renewable energy also include reduced energy supply prices and a decrease in the levelized cost of heat and electricity [18], [19], [20]. In today's world, there are many types of renewable energy. Some forms of renewable energy that we have today are hydropower, energy meteorology, solar energy, biomass, storage technology and wind energy. Due to its rising popularity and request from some companies and development organizations during the second half of the Eighties, the Postgraduate Programme Renewable Energy (PPRE) was founded [9]. In this paper, the researcher focuses on solar energy. Figure 2.1 shows where renewable energy is growing.

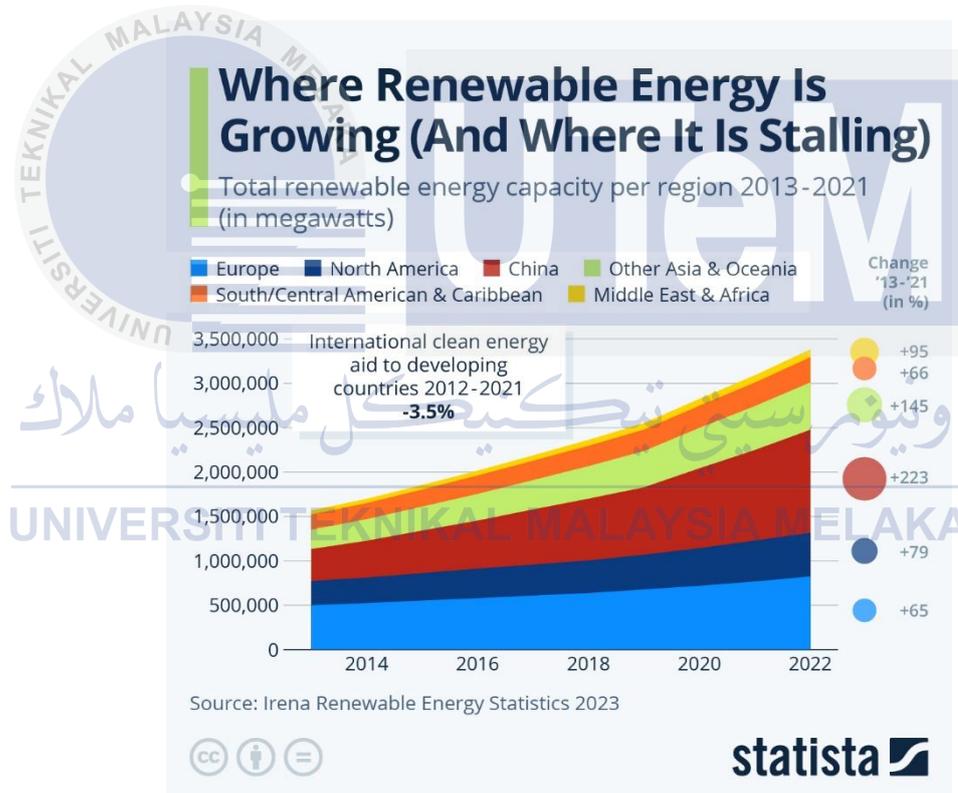


Figure 2.1 The growth of renewable energy

2.2 Solar Energy

The increasing depletion of fossil fuels, which account for over 80% of global energy usage, necessitates immediate research into sustainable and renewable energy alternatives. Recent years have seen a sharp rise in the demand for clean, environmentally friendly energy, which has raised energy prices globally [21]. The need for this renewable energy drives our efforts to create greener energy sources and, soon, to decarbonize the earth. [22], [23]. The use of renewable energy as a future energy supply is gaining a lot of interest around the world. Solar energy is a widely available, sustainable, and renewable source of energy [1]. Renewable energy technologies are alluring energy sources that are clean, friendly to the environment, and help meet the world's energy needs [24], [25], [26]. Applications for renewable energy include solar, wind, biomass, and geothermal energy. Solar energy is the most widely used application technology, followed by wind energy [27], [28].

As a renewable energy source, solar energy (including concentrating solar power (CSP) and solar photovoltaic (PV) power) contributes 3.6% to global electricity output. However, it has firmly established itself among other renewable energy technologies, accounting for almost 31% of total installed renewable energy capacity in 2022. Solar energy, with an installed capacity of 1053 GW in 2022, is the second largest installed renewable energy technology, trailing only hydroelectric technology, which has 1392 GW [2].

Concentrated solar power (CSP) technology harnesses the sun's rays to heat a liquid and produce steam in a limited area by the use of mirrors or lenses [29]. After that, the steam turns a turbine to generate energy. This method allows for the classification of CSP systems into three groups: power towers, parabolic troughs, and dish-stirling systems [30], [31]. The key is solar PV technology, which is the most promising clean and renewable energy source due to its cheap maintenance costs, ease of installation, dependability, and lack of fuel requirements [32]. Furthermore, the photovoltaic modules offer major advantages in clean energy production due to their low cost and lack of wear and noise [33]. Since photovoltaic solar technology has advanced so much in recent years, solar PV capacity has been installed all over the world to meet demand for electricity [27], [28]. As a result, solar energy has a huge chance to contribute significantly to the switch to cleaner, more sustainable energy sources. Additionally, solar energy initiatives provide an affordable and reliable source of

electricity. Table 2.1 summarises solar energy classifications by energy induced, applications, and efficiency. Figure 2.2 shows the yearly world solar photovoltaic estimated deployments from 2000–2050 [34].

Table 2.1 Lists of solar energy technology classification

No.	Technology	Applications	Induced energy	Range of efficiency	Refs.
1	PV	Solar PV solar water pumping	Powering devices	20-40%	[75-77]
2		Solar Concentrating PV	Heat/Electricity	33-47%	[78, 79]
3		Solar PV desalination	Powering devices	-	[80, 81]
4		Solar PV cathodic protection	Powering devices	-	[82]
5		Solar PV electrification	Electrification	15-20%	[78]
6		Solar PV vehicle	Powering devices	30 %	[83]
7		Solar Photovoltaic /thermal	Heat/Electricity	60-80%	[84]
9	CSP	Solar Central receiver	Heat	42-45%	[85]
10		Solar Parabolic dish	Heat	40%	[86]
11		Solar Parabolic trough	Heat	52-55%	[87]
12		Solar Liner Fresnel reflector	Heat	37%	[88, 89]
13		Solar cooker	Heat	4-7%	[90]
14		Solar dehydration	Heat	20- 60%	[91]

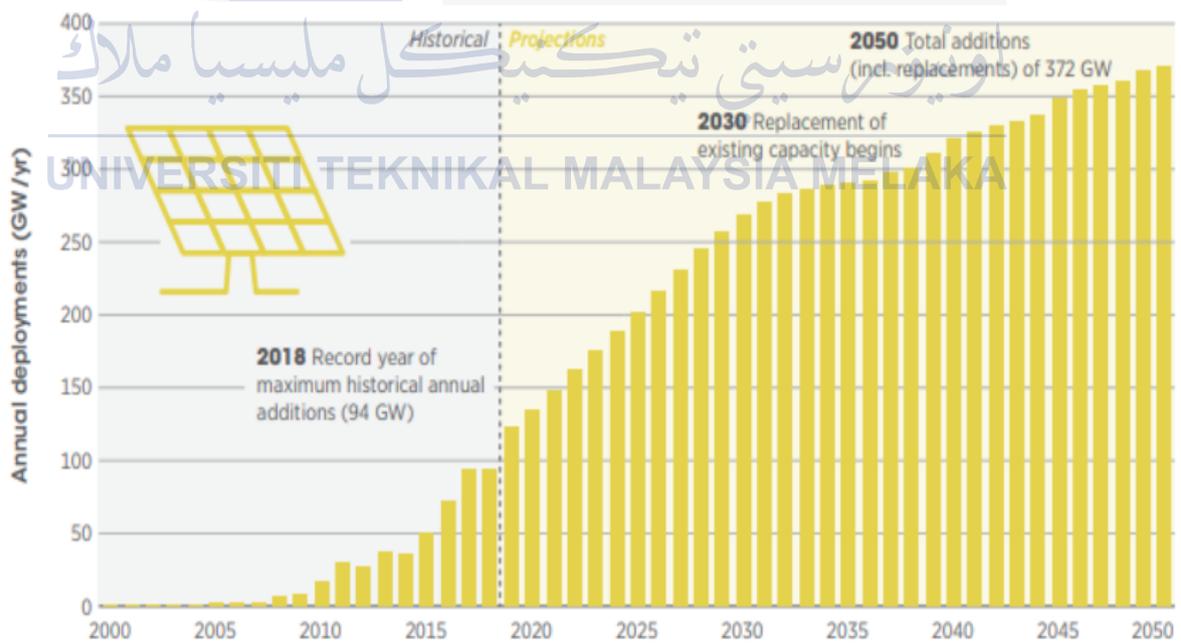


Figure 2.2 Yearly world solar photovoltaic estimated deployments from 2000–2050 [64]

2.3 Solar Photovoltaic (PV) Power

Photovoltaic (PV) solar energy is generated by converting sunlight (solar radiation) into electricity using solar panels, a photoelectric effect-based technology. Photovoltaic cells in solar panels convert sunlight into direct current (DC) power, which is subsequently transformed into alternating current (AC) electricity for usage in homes and businesses. Monocrystalline, polycrystalline, amorphous silicon, and other thin-film semiconductor materials can be used to make solar cells. Because it is modular, it can be utilized in installations ranging from massive ground-mounted solar plants to small roof panels. As a result of this characteristic, solar PV has emerged as the major type of solar energy in use in recent years, accounting for more than 95% of all installations [3].

Tables 2.2 and 2.3 show the global installed solar PV capacity during the last 10 years, as well as the contributions of the top fourteen nations. In the early years of solar PV development, Europe was the largest donor to worldwide solar PV projects. As seen in Table 2.2, this continent accounted for 60% of the world's solar PV installations in 2013. Since 2013, rapid solar PV development has occurred in various areas, particularly in China. In 2017, China surpassed Europe as the largest solar PV market, accounting for around one-third of global installed capacity. In 2022, the world's cumulative installed solar PV generating capacity will surpass 1046 GW. Table 2.3 illustrates a massive growth in solar PV installed capacity of around 22% (192 GW) between 2021 and 2022. While the top three installers are China, the United States, and Japan, China's relative contribution accounts for roughly 37% of total solar PV installation in 2022. The most substantial growth in the solar PV industry happened in China, the United States, and India in 2022, with increases of 86.1 GW, 17.8 GW, and 13.5 GW, respectively [2].

Figure 2.3 shows the contribution of each continent in the world's solar PV installed capacity in 2018, followed by 2030 and 2050 based on IRENA's REmap analysis. In comparison to the PV installations in 2018 (481 GW), the world's PV installed capacity is projected to increase almost six times by 2030 (to 2841 GW) and almost 18 times by 2050 (to 8519 GW, of which the distributed scale (rooftop) would account for 40% while the remaining 60% would be utility scale). Asia will proceed to lead the solar PV market by about 65% of the world's PV installations (mainly China with 76% of the total), followed

by North America at 15% (primarily the US with over 90% of the total) and Europe at 10% by 2030. By 2050, Asia, primarily China, is expected to maintain its leadership in the solar PV market with 4837 GW (about 57% of the world's PV installations), followed by North America at 21% and Europe at 11%. Meanwhile, a much larger market growth is anticipated for both Africa and South America by 2050, as shown in Figure 2.3. By 2030, Asia will have over 65% of the world's PV installations (mostly China with 76% of the total), followed by North America at 15% (largely the US with over 90% of the total) and Europe at 10%. Asia, particularly China, is predicted to maintain its leadership in the solar PV market by 2050, with 4837 GW (about 57% of the world's PV installations), followed by North America (21%), and Europe (11%). Meanwhile, as illustrated in Figure 2.3, Africa and South America are expected to have substantially bigger market growth by 2050 [1].

Figure 2.4 depicts the evolution of the global weighted-average total installed cost of solar PV plants from 2014 to 2050. The global weighted-average total installed cost of solar PV projects decreased by almost 67% from 2652 USD/kW in 2014 to 876 USD/kW in 2022. The 2022 weighted-average total installed cost was recently reduced by around 4% as compared to the 2021 number. According to the findings of IRENA's REmap analysis, the global weighted-average total installed cost of solar PV projects would fall from 876 USD/kW in 2022 to 340-834 USD/kW by 2030 and 165-481 USD/kW by 2050 [1].

Figure 2.5 depicts the difference in total installed cost trends for solar PV projects in fifteen main markets from 2010 to 2022. The country-weighted average total installed cost dropped from 2010 to 2022 in the top fifteen markets, with India experiencing the greatest reduction (89%) and Germany experiencing the smallest reduction (76%) in total installed cost. The reduction in total installed cost in 2022 compared to 2021 values ranged from 22% in Chile to 4% in the US, while the increase in total installed cost in 2022 compared to 2021 values ranged from 34% in Germany and France to 2% in India. India had the lowest total installed cost among the top fifteen markets in 2022, at 640 USD/kW, followed by Turkey (690 USD/kW), China (715 USD/kW), Italy (771 USD/kW), and Spain (778 USD/kW). Japan had the highest 2022 total installed cost among the aforementioned fifteen major markets, at 1905 USD/kW, followed by the Netherlands (1221 USD/kW) and France (1157 USD/kW) [1].

Since 2010, the global weighted-average LCOE for solar PV plants has varied as seen in Figure 2.6. The global weighted-average LCOE of solar PV technology has been decreased by approximately 89%, from 0.445 USD/kWh in 2010 to 0.049 USD/kWh in 2022. Since 2014, the LCOE of PV technology has fallen into the range of fossil fuel electricity costs. The most recent global weighted-average LCOE reduction from 2021 to 2022 was roughly 3%. According to the findings of IRENA's REmap analysis, the LCOE for solar PV projects would fall from 0.049 USD/kWh in 2022 to an average of 0.02-0.08 USD/kWh by 2030 and 0.014-0.05 USD/kWh by 2050 [1].

Since 2010, Figure 2.7 depicts the regional weighted-average LCOE of solar PV projects in the top fifteen markets. The country weighted average LCOE decreased in the top fifteen markets between 2010 and 2022, with Australia having the highest and lowest decreases (91%) and the United States having the lowest (75%). The reduction in the 2022 LCOE relative to the 2021 estimates ranged from 9% in Chile and Australia to 1% in the United States. Meanwhile, the rise in 2022 LCOE over 2021 values ranged from 27% in Germany and Mexico to 2% in India. The observed countries weighted-average LCOE in 2022 was within 0.037-0.1 USD/kWh [1].

Figure 2.8 depicts the change in global weighted-average capacity factor for solar PV plants between 2010 and 2022. From 13.8% in 2010 to 16.9% in 2022, there has been a trend towards greater capacity factors. The observed increase in capacity factor is primarily attributable to three essential factors: 1) system performance enhancement through loss reduction, 2) use of solar tracking devices, and 3) preference for deployment in areas with greater radiation levels [1].

Table 2.2 Global installed solar PV capacity from 2013 to 2022.

	Solar PV capacity (MW)									
	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
World	136 572	176 113	224 070	296 112	390 878	483 495	585 868	713 918	855 162	1 046 614
Africa	651	1 544	1 917	3 030	4 675	7 165	8 408	9 734	10 543	11 556
Asia	36 055	60 346	90 236	140 125	211 488	275 827	332 111	409 433	484 496	596 530
Europe	81 878	88 783	97 292	103 861	109 987	119 291	139 951	160 474	187 822	225 478
N. America	12 358	18 463	25 285	36 973	46 070	55 890	67 881	84 728	105 695	124 946
S. America	198	465	921	1 589	3 672	5 512	8 562	13 164	20 687	32 665
Oceania	4 607	5 355	6 076	6 857	7 574	8 878	13 290	18 354	23 339	27 397

Table 2.3 Top fourteen solar PV energy installers in 2022.

N/s	Country	Installed capacity (GW)
1	China	392.4
2	USA	111.5
3	Japan	78.8
4	Germany	66.5
5	India	62.8
6	Australia	26.8
7	Italy	25.1
8	Brazil	24.1
9	Netherlands	22.6
10	South Korea	20.9
11	Viet Nam	18.5
12	Spain	18.2
13	France	17.4
14	UK	14.4

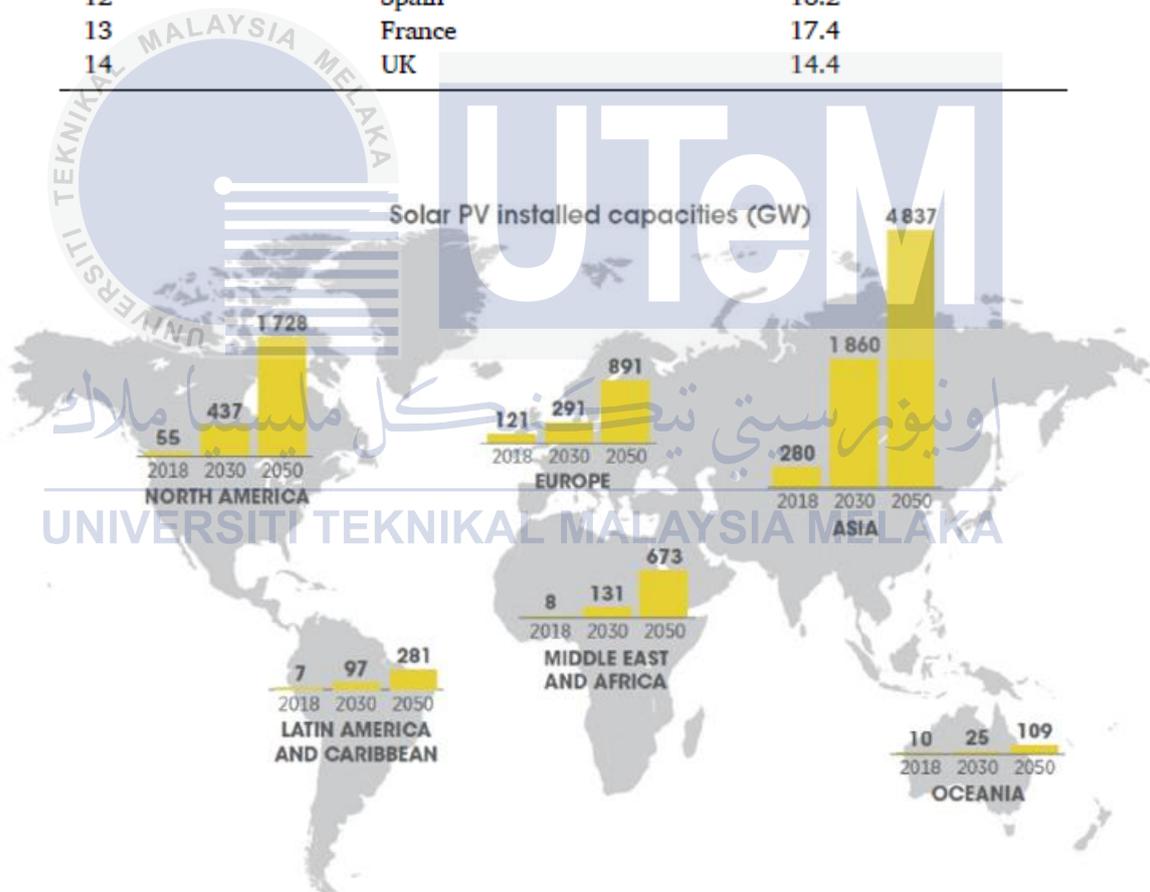


Figure 2.3 The world solar PV installed capacity by 2050.

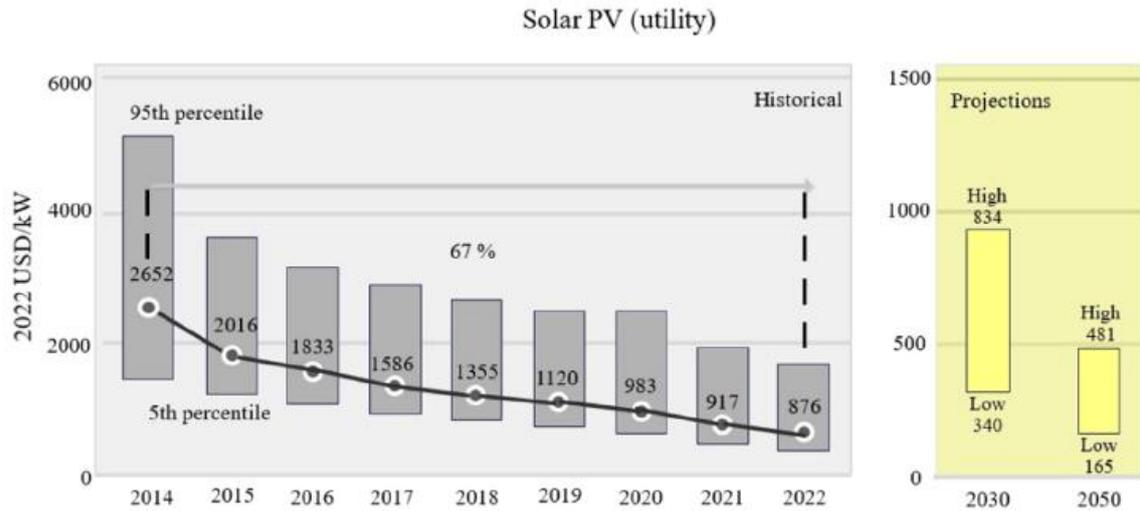
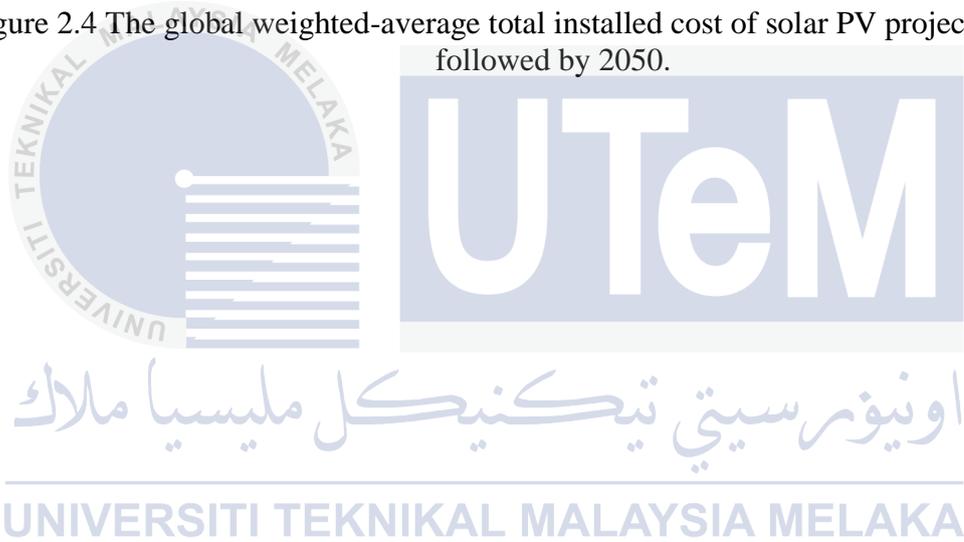


Figure 2.4 The global weighted-average total installed cost of solar PV projects since 2014, followed by 2050.



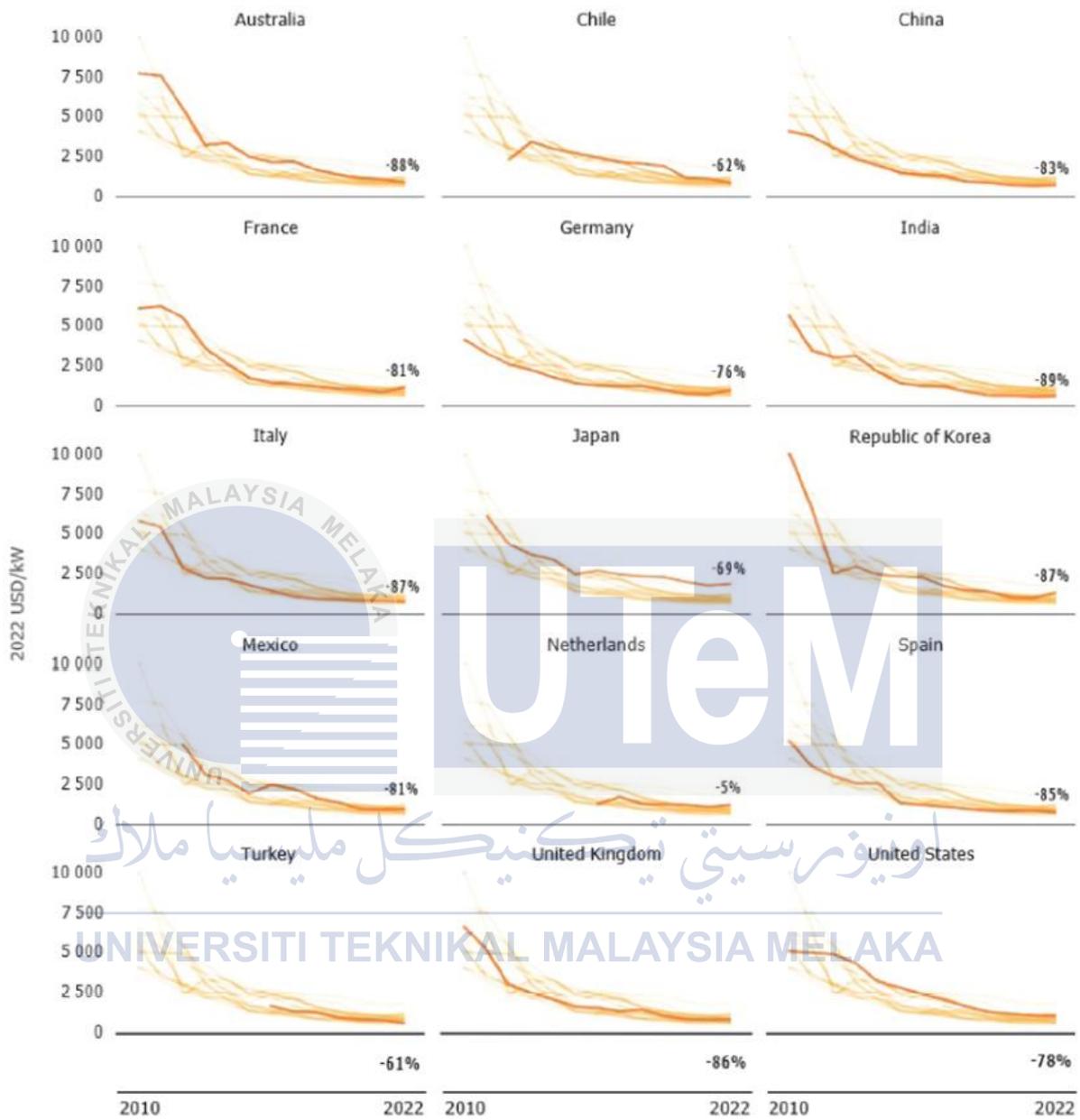


Figure 2.5 The total installed cost trends for solar PV projects in major markets since 2010.

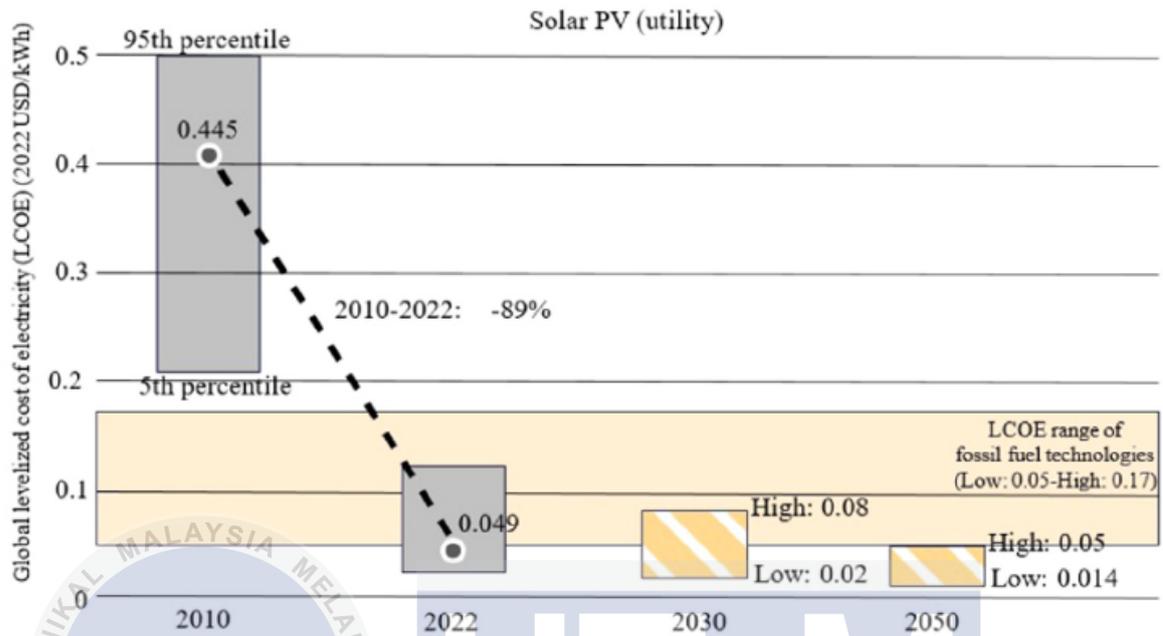


Figure 2.6 The global levelized cost of electricity for solar PV projects since 2010, followed by 2050.

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

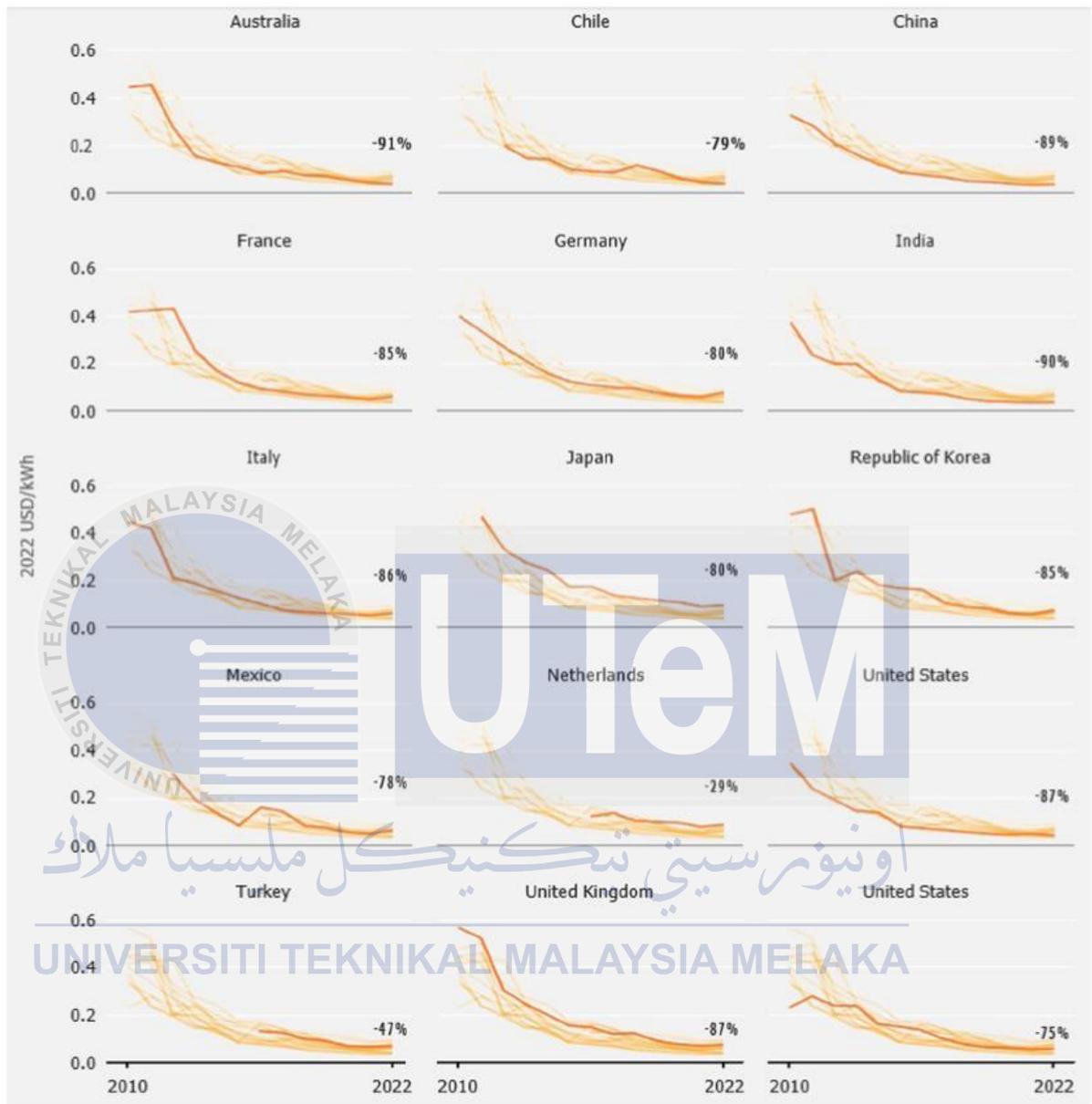


Figure 2.7 Regional weighted average levelized cost of electricity for solar PV projects in the major markets since 2010.

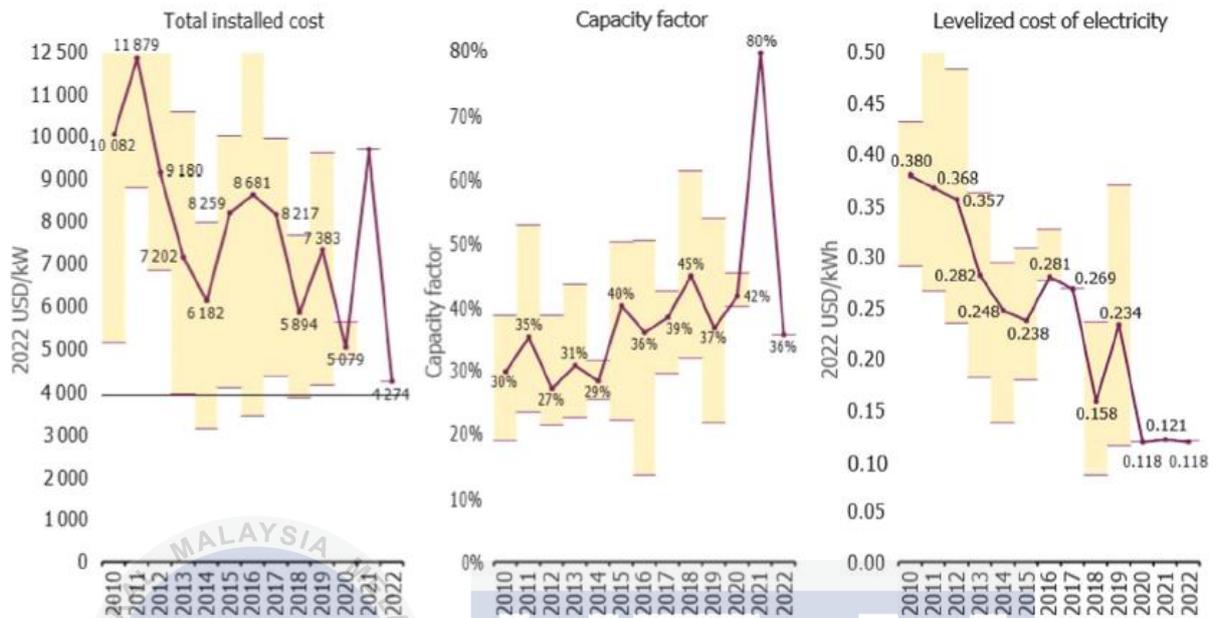


Figure 2.8 The global weighted average LCOE, capacity factor, and total installed costs for CSP projects between 2010 and 2022.

2.4 Defects on Photovoltaic Panels

Solar energy has garnered a lot of attention in recent years as a feasible replacement for fossil fuels. It is a renewable and sustainable energy source. Nonetheless, various faults, such as cells, diodes, or multiple cells and multiple diodes, may have an impact on the effectiveness and performance of solar panels. These defects can considerably impair energy generation; hence it is vital to discover and categories them as soon as possible [4].

When examining solar monitoring data or performing visual inspections, there are various indicators that solar panels are underperforming. A large decline in energy production relative to the rated capacity of the PV system plainly indicates underperformance. Regularly monitor solar energy production and compare it to the projected output. The collection of dust, grime, or debris on the panels might diminish efficiency. Shade from neighboring trees or buildings can also cast shadows on the panels, reducing their performance. Cracks, cracks, and other obvious damage to the surface of the panels might impair their ability to convert sunlight into electricity. Electrical issues with the system, such as loose connections, broken inverters, or malfunctioning wiring, can also result in decreased performance [5].

2.4.1 Snail Trails

Snail trails are a sort of fault that can be observed with the naked eye as dark or discolored patterns on the surface of solar panels. They are created by a chemical reaction caused by moisture and oxygen within the panel's encapsulation material. Over time, this process might result in the creation of conductive channels, lowering the overall efficiency of the panel. In order to detect snail trails, regularly inspect the surface of the panels for any noticeable dark streaks or deterioration. Under some lighting circumstances, such as direct sunlight, snail trails are more visible [5]. Figure 2.9 shows snail trails defect on a photovoltaic panel.

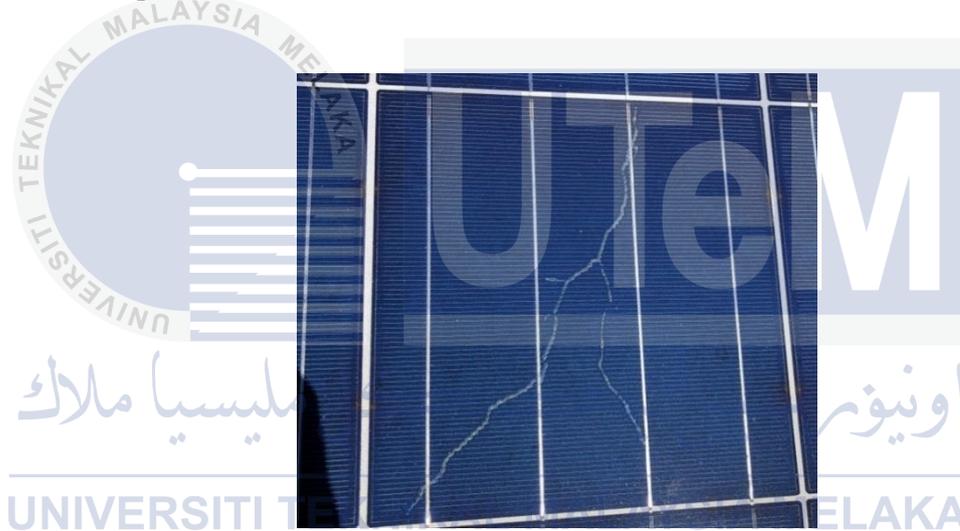


Figure 2.9 Snail trails defect on a photovoltaic panel.

2.4.2 Hotspots

Hotspots form when certain cells inside a solar panel overheat as a result of localized shadowing, dirt, or manufacturing flaws. These hotspots might cause irreversible damage to the damaged cells and lower the panel's overall output. Use thermal imaging during the day, when the panels are in direct sunlight, to find hotspots. Hotspots are parts of the panel that have significantly greater temperatures than the rest of the panel [5]. Figure 2.10 shows hotspots defect on a photovoltaic panel. Figure 2.11 and Figure 2.12 shows hotspots defect on a photovoltaic panel under thermal imaging.

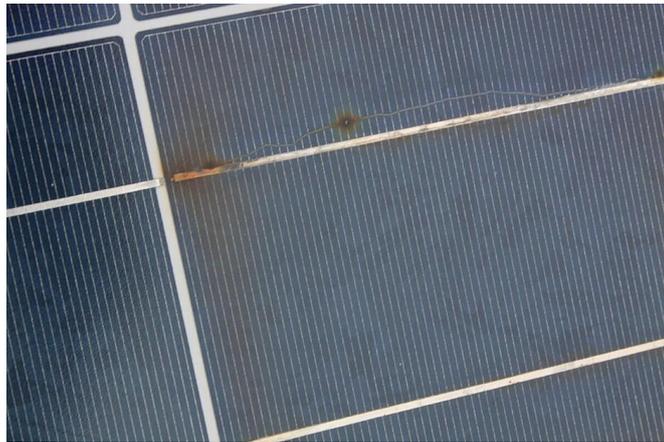


Figure 2.10 Hotspots defect on a photovoltaic panel.

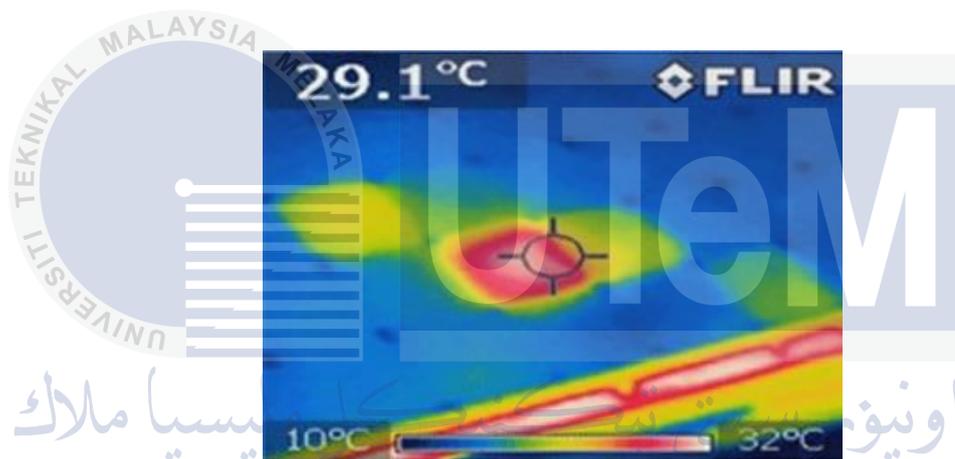


Figure 2.11 Hotspots defect on a photovoltaic panel under thermal imaging.

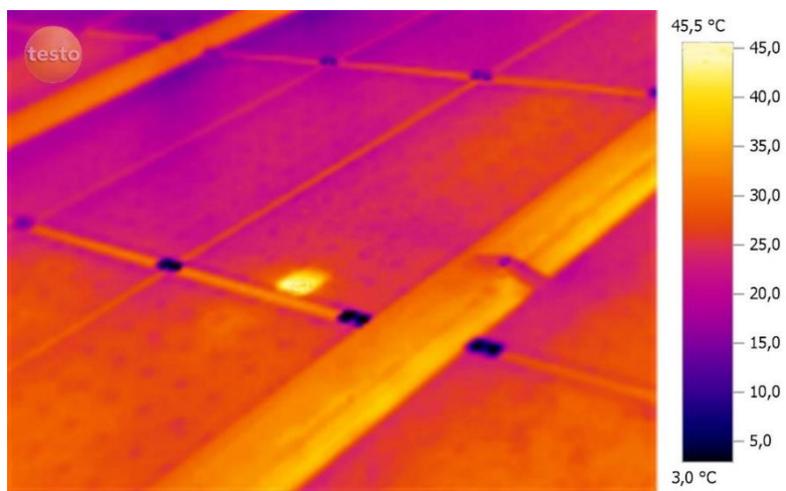


Figure 2.12 Hotspots defect on a photovoltaic panel under thermal imaging.

2.4.3 Shading and Shadowing

Shading and shadowing are typical problems that can have a negative impact on the performance of solar panels. Even partial shade on a single cell can significantly reduce the overall energy output of the panel. During the day, inspect the panels for shade and shadowing difficulties, especially when the sun is at a low angle. Look for shadows cast by objects, surrounding structures, or plants on the panels. Homeowners and business owners can keep an eye on this as they spend more time around the panels; make sure they know to call their installer if they see excessive shading or shadowing. Analyze the system's monitoring data as well for any odd dips in energy production [5]. Figure 2.13 shows shading and shadowing defect on photovoltaic panels.



Figure 2.13 Shading and shadowing defect on photovoltaic panels.

2.4.4 Microcracks

Microcracks, also known as microfractures, are microscopic fissures found in photovoltaic cells. Mechanical stress during installation, shipping, or environmental conditions such as temperature variations are common causes of this sort of solar damage. These microcracks can impair panel performance. Visually inspect the panel's surface under adequate lighting to detect microcracks; cracks may show as faint lines on the cells or the

surrounding material. You can also utilize electroluminescence crack detection (ELCD) testing, which detects microcracks that a visual inspection may miss [5]. Figure 2.14 shows microcracks defect on photovoltaic panels.

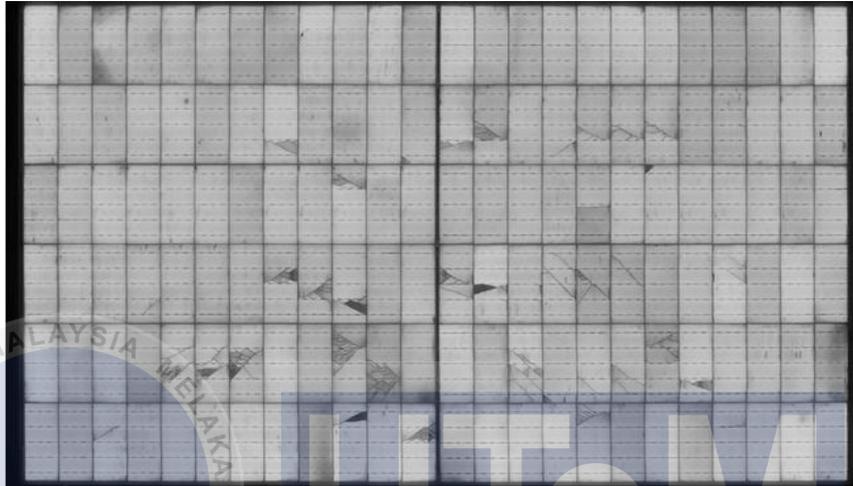


Figure 2.14 Microcracks defect on photovoltaic panels.

2.4.5 Delamination

PV panel delamination is a major problem that happens when the layers of materials within the PV module separate or become separated. It can occur as a result of moisture entering the back sheet through fractures, resulting in a drop in panel efficiency and busbar corrosion. Conduct a thorough visual inspection of the solar panels to detect delamination. Look for bubbles, blisters, or separations between the panel's layers, as well as discoloration or dark areas on the panel's surface. Electroluminescence (EL) testing, which captures images of the panel in the dark, can also reveal delamination [5]. Figure 2.15 shows delamination defect on a photovoltaic panel.



Figure 2.15 Delamination defect on a photovoltaic panel.

2.4.6 Potential Induced Degradation (PID)

Potential Induced Degradation PID is a phenomenon that impairs the performance of solar panels as a result of a high voltage potential difference between the solar cells and the frame, glass, or mount. This potential difference can cause solar cells to degrade, resulting in lower energy output. To detect PID, keep a close eye on the functioning of the solar panels and look for symptoms of declining efficiency. PID symptoms include a quick decrease in energy output or a considerable drop in performance when exposed to high humidity and high temperatures [8]. Figure 2.16 shows before and after PID defect on a photovoltaic panel. Dark cells represent PID susceptible cells [35].

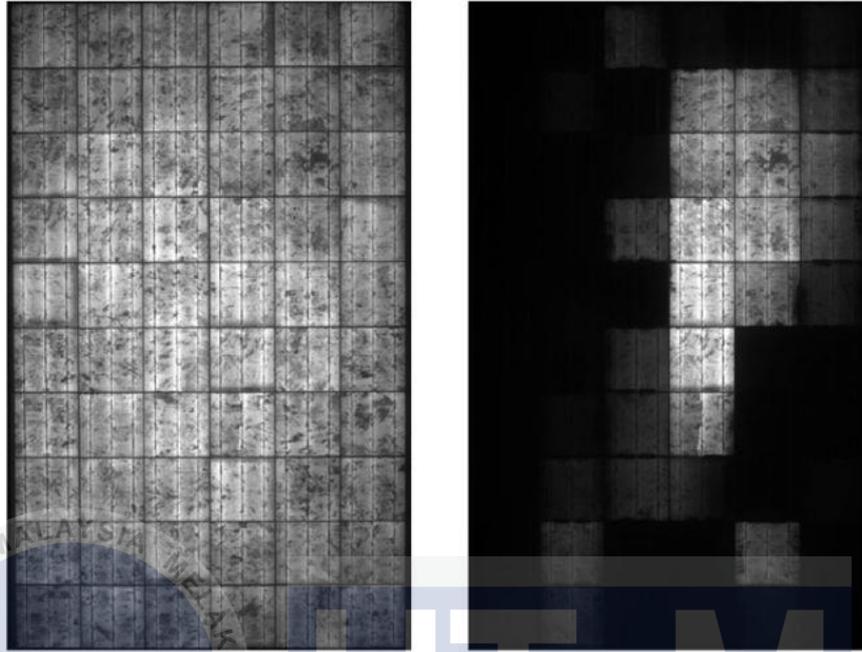


Figure 2.16 Before and after PID defect on a photovoltaic panel.

2.4.7 Diode Failure

— Diodes are critical components in solar panels that prevent reverse current flow. Short-circuited bypass diodes cause power losses of 33% or more and can cause hotspots. To detect diode failure, check the system's performance on a regular basis, looking for any substantial dips in energy generation or unexpected patterns in the monitoring data. Inspect the panels for evident signs of damage or discoloration as well [5]. Figure 2.17 shows diode failure defect on a photovoltaic panel.

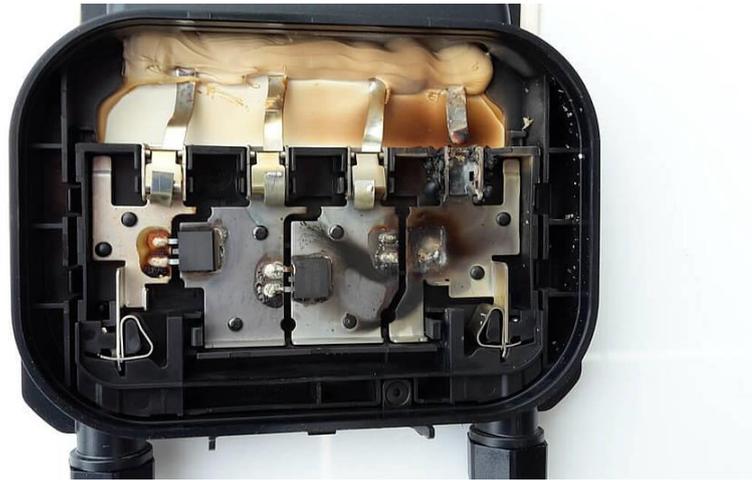


Figure 2. 17 Diode failure defect on a photovoltaic panel .

2.5 Defect Detection

Renewable energy generation has been expanding at an exponential rate every year. The capacity of photovoltaic (PV) solar power facilities has increased by roughly 28% every year on average. The rise in capacity is primarily due to improvements in photovoltaic panel efficiency and the size of these solar power facilities. As the capacity of solar power facilities has increased in recent years, elaborate systems to manage their maintenance and operations have become necessary. PV modules degrade over time due to a variety of internal and external variables, resulting in power output as low as 50% for certain severe flaws. As a result, these flaws must be identified as soon as feasible to stop further deterioration. Faults in PV modules are issues that diminish power output and can be characterized as either permanent (electrical disconnection, wiring losses, and ageing) or transient (dust, shadow, and bird droppings). It is therefore critical to not only identify the defect but also to determine the type of fault so that suitable steps can be implemented [36].

For monitoring purposes, a lot of solar power plants currently use manual inspection procedures [36]. Traditional solar panel evaluation involves staff visiting the solar park and visually assessing each panel. This procedure is time-consuming and frequently results in errors [4]. Power stations with millions of PV modules distributed across a vast region cannot use this technology. These power plants have very high operating and maintenance costs; thus, we need an automated system that can lower these costs and increase system efficiency. Techniques based on electrical characteristics, Infrared imaging, visual

inspection, ultrasonic inspection, electroluminescence imaging, and lock-in thermography are among the general categories of existing fault analysis techniques. Because thermal imaging is readily accessible and applicable to large solar plants, we employ it in our work to identify problems in solar power facilities [36]. Automation of this process is now possible because to advancements in deep learning and computer vision technology, which will make it faster, more effective, and less expensive [4].

2.5.1 Previous Work on Defect Detection

This section addresses earlier attempts to discover and characterize flaws in solar panels, as well as their limitations. A thermography-based approach for detecting defects and faults in PV systems was developed by Aghaei et al. [37]. Their program discovered infrared photos with a hot temperature zone. To reduce noise, the original image was transformed to grayscale before using Gaussian filtering. After that, the images were processed to a binary model to distinguish between hot and cool regions in the PV modules. Finally, the Laplacian model was utilized to describe the features of problematic components and evaluate the boundary region of the panel.

Jaffery et al. [38] devised an approach for detecting faults using infrared technology. The diagnosis system employs fuzzy logic for intelligent and automatic detection. A significant difference in colour pattern between faulty and healthy panels is used to classify different defects. The knowledge for classification is used to define the fuzzy rule basis. Nonetheless, this article only provides the type of problem and not the location or region of the fault.

Dunderdale et al. [39] used a scale invariant feature transform (SIFT) feature descriptor, spatial pyramid matching, and deep learning to recognize and classify thermal infrared images of PV modules. SIFT descriptor is used in conjunction with the random forest model and SVM kernels, specifically the polynomial and radial basis functions, to detect defects. In this 2-class classification challenge, random forest is found to perform better. Defect classification is performed utilizing five classes and methodologies such as the bag of visual words model, spatial pyramid matching, and deep learning. Deep learning models based on VGG-16 and MobileNet outperformed feature-based techniques. The

implemented models have the best flaw detection accuracy of 91.2% and classification accuracy of 89.5%. Even however, not enough data have been used in this investigation, and the location of the issue has not been determined, which could have simplified the maintenance tasks.

Hazem Munawer Al-Otum created an effective fault detection and classification strategy employing multi-scale convolutional neural networks (CNN) based models in two scenarios: a) a transfer learning-based approach based on two selected deep neural networks (DNN) (ResNet18 and ShuffleNet) and b) an independent light-depth CNN (denoted as CNN-ILD). CNN-ILD is a lightweight CNN that employs parallel convolutional branches with low-to-high kernel sizes. CNN-ILD was able to capture a wide range of features, from basic to particular textures and patterns, thanks to this design. Following proper preprocessing and class categorization, the public ELPV dataset was used. The experimental results showed promising classification results of PV cell abnormalities in electroluminescence pictures (88.41%-98.05%). CNN-ILD has an advantage over the other two proposed pretrained DNNs in terms of computational power/time and stability [40]. Figure 2.18 shows electroluminescence imaging setup. Figure 2.19 shows examples on EL images taken from the ELPV dataset: a) mono c-Si cells (mc-Si), and b) poly c-Si cells (pc-Si).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

B Sandeep, D Saiteja Reddy, Aswin R and R Mahalakshmi propose a way to monitor PV modules and hotspots detection using TensorFlow. This method consists of creating hotspot images that are comparable to real solar panel photographs, training the model, verifying it with validating images, and then testing images in which the model determines the hotspot kind and classification. The model is trained using several hotspot image kinds, and once each generated image has been validated, it can identify and categories the hotspot type, indicating that the hotspot classification is complete. By identifying the type of hotspot, the appropriate action can be done right away to prevent panel damage. The results of this method shows that images must be fed into the model in order for it to classify them. This can be done by uploading images to Python Tkinter and identifying the type of hotspot. Because this model can also identify the sort of hotspot, it was able to detect the colour of the hotspot and classify it. Figure 2.20 shows the image classification result. Using the generated data, the model was tested and trained. Figure 2.21 displays the accuracy results

from training and validation, whereas Figure 2.22 displays the loss. As a result of the training and validation accuracy improving linearly for each epoch, the model's accuracy in identifying hotspots is improved. It is clear from the loss graph that each epoch's loss is reducing linearly. The validation loss graph does not show the training loss spike that occurs at one point in time. Consequently, precision is enhanced [41].

Jing Huang, Keyao Zeng, Zijun Zhang and Wanhan Zhong develop a solar panel defect detection design based on YOLO v5 algorithm. To compensate for the low detection efficiency of the conventional defect detection techniques, the YOLO v5 algorithm is enhanced. To improve the LCA attention mechanism, it is first improved based on coordinate attention to obtain a larger target range, which can improve the sensing range of target features in addition to fully capturing feature information; second, the feature information with excessive pixel differences is balanced by assigning different weights to the weighted bidirectional feature pyramid, which is more conducive to multi-scale fast fusion of features; and finally, the typical coupled head of the YOLO series is replaced with a decoupled head, which can improve the task branch and detection accuracy. Comparative studies conducted on the solar panel defect detection data set reveal that following algorithmic improvements, the mAP is up to 95.5%, 2.5% higher than pre-improvement levels, and the recall rate increases by 2.4% and 1.5% overall. It can more precisely identify any flaws, standardise solar panel quality, and guarantee electrical safety [42]. Figure 2.23 shows the YOLO v5 network structures.

In this paper, the researcher decided to use YOLO v8 as the model to detect defect on PV modules. The researcher believes that YOLO v8 will provide a higher accuracy result. The researcher will use thermal imaging sensor to detect the temperature attributes of the PV modules. The model and the thermal imaging sensor will be attach to a drone to allow human to detect defects in one place. Table 2.4 summarize the previous work on defect detection.

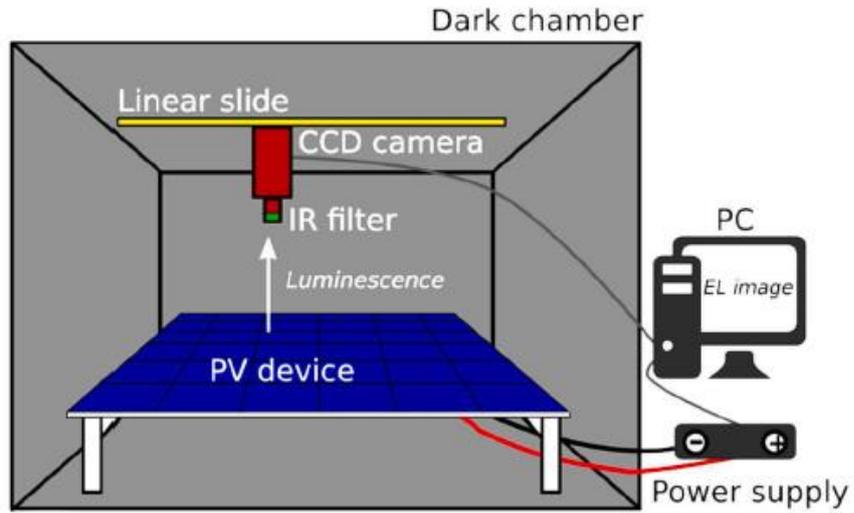


Figure 2.18 EL imaging setup.

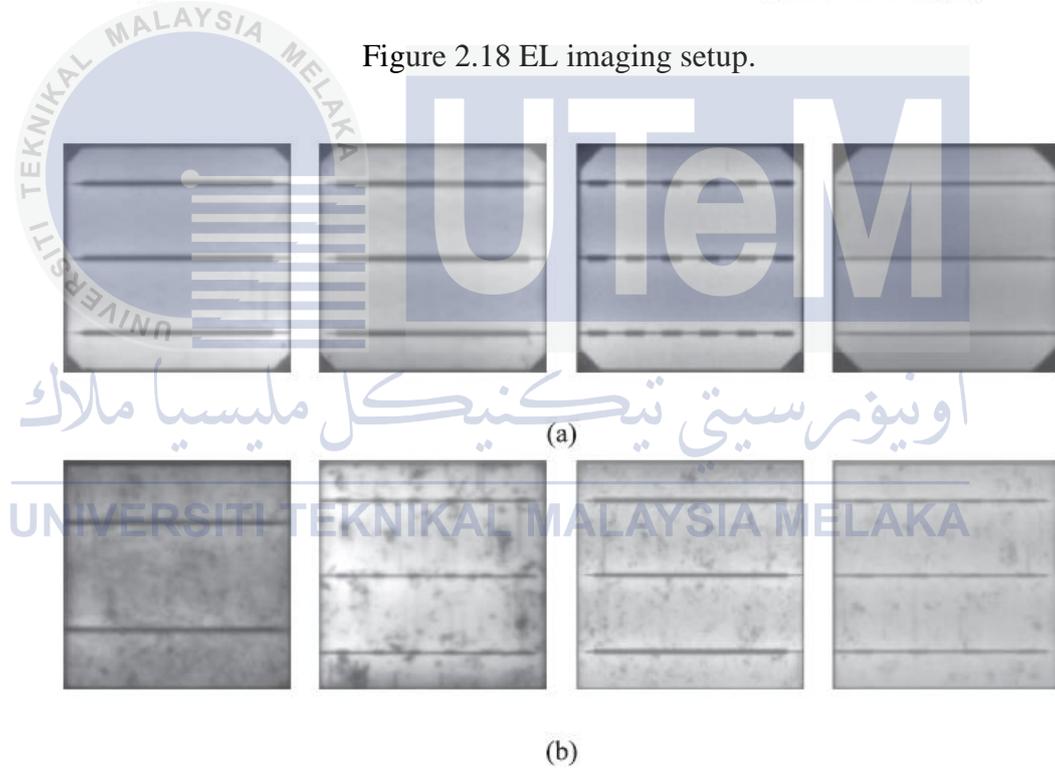


Figure 2.19 Examples on EL images taken from the ELPV dataset: a) mono c-Si cells (mc-Si), and b) poly c-Si cells (pc-Si).

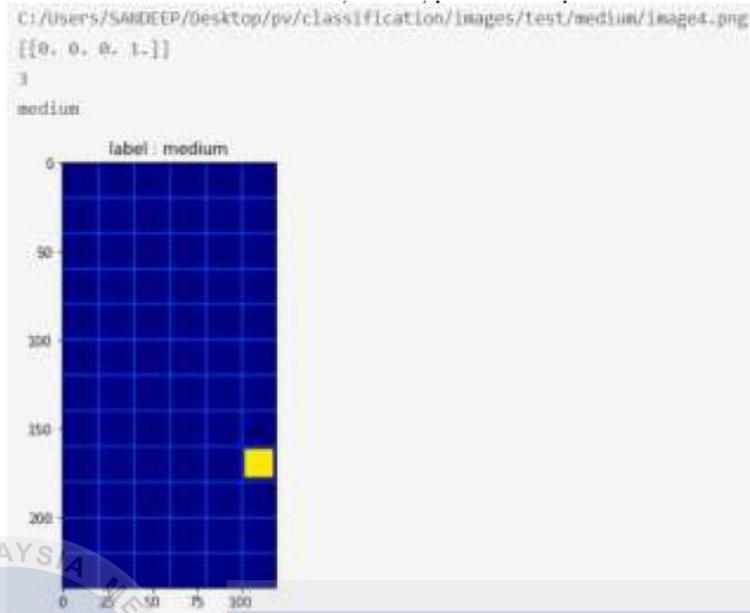


Figure 2.20 Image classification result.

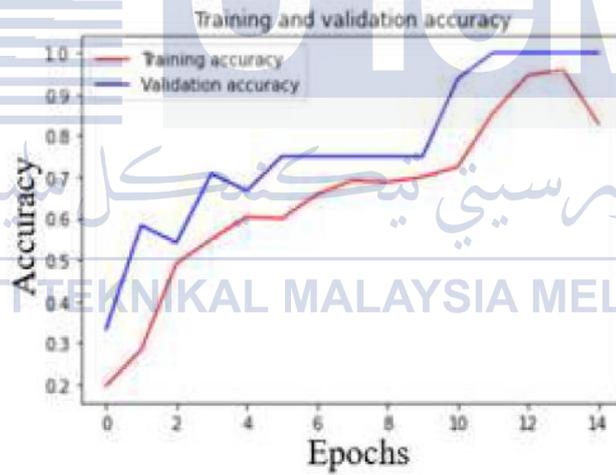


Figure 2.21 Accuracy results.

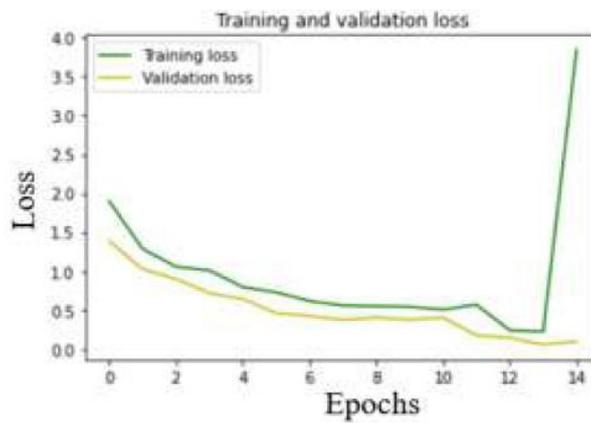


Figure 2.22 Loss results.

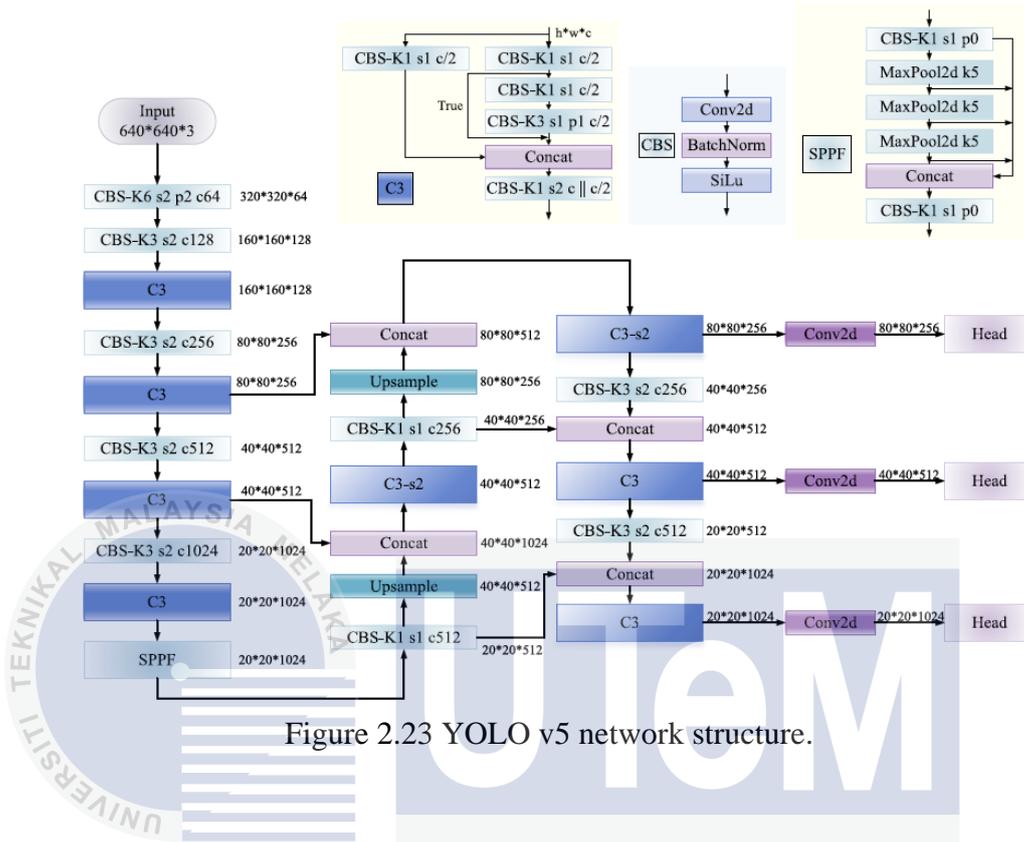


Figure 2.23 YOLO v5 network structure.

Table 2.4 Comparison of previous work.

Researcher/s	Type of Technology Used to Detect Defect	AI Model
Jaffery [38]	Infrared technology	Fuzzy logic
Dunderdale [39]	Thermal infrared	VGG-16 and MobileNet
Hazem Munawer Al-Otun [40]	Electroluminescence	ResNet18, ShuffleNet and light-depth CNN
B Sandeep, D Saiteja Reddy, Aswin R and R Mahalakshmi [41]	Hotspot images	TensorFlow
Jing Huang, Keyao Zeng, Zijun Zhang and Wanhan Zhong [42]	Excessive pixel differences	YOLO v5

2.6 Artificial Intelligence

Artificial intelligence (AI) is a method that refers to a system or a machine that imitates human intelligence in order to accomplish real-world operations. AI enables the system to be trained from data and to reason and learn from experience in order to address specific challenges. Based on the facts, it can heuristically refine itself. AI applications include enhanced web search engines, self-driving cars, games, human speech recognition, recommendation systems, and healthcare, among others [43].

AI was developed in the computer science sector around 1950, and it mimicked the human mind to create machines that can analyze, methodize, and perform based on the data presented to the system, which will be advantageous when massive amounts of datasets are employed. AI machines are widely employed in the industrial domain, prompting greater research in engineering domains such as NLP (natural language processing), disease detection and medical, and science. AI computers used to learn from their previous experiences, which was useful in problem solving, and it has been applied in several application areas to improve AI machine performance. Figure 2.24 depicts the relationship between Artificial Intelligence, Machine Learning, Deep Learning, and Explainable Artificial Intelligence [43].

Machine Learning (ML) is a technology that allows a computer to recognize patterns, make more accurate predictions, and perfect itself through experience without being explicitly programmed to do so. Machine Learning is utilized to create an AI-powered application. The ML Methodologies are used in this procedure. The ML procedure is depicted in Figure 2.25. AI is frequently utilized to make choices. When combined with AI, the system can do jobs more quickly and predict the judgements required to handle complex problems, assess risks, and evaluate company success [43].

Deep learning (DL) is the study of algorithms that are impacted by the structure and function of the human brain. DL use artificial neural networks to build an intelligent model and tackle complex issues. To train a model, DL uses both structured and unstructured data (e.g., visual assistants like Siri, Alexa, and facial recognition, among others). DL is utilized in medical research and the prognosis of potentially fatal diseases. Deep Neural Networks (DNNs) have recently demonstrated exceptional prediction performance [43].

Deep learning has made significant progress in recent years as a result of its improved computing power and ability to deliver a better solution for a wider number of datasets. Deep Learning (DL) is a subtype of AI that is generated using an artificial neural network. The input data in this DL procedure will be trained by themselves over mathematical illustration. Convolutional Neural Networks (CNN), Visual Geometric Group Net (VGGNet), Residual Network (ResNet), Fully Convolutional Networks (FCNs), U-net, Deep feed forward networks, Siamese Neural Networks, and Graph Neural Networks are some DL models [17].

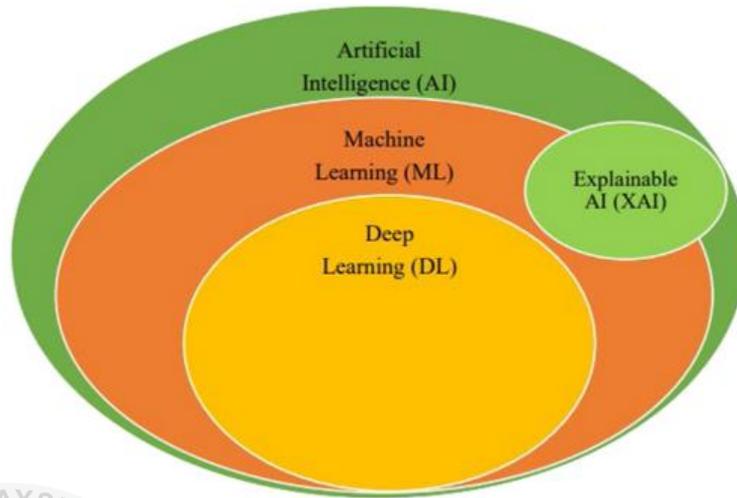


Figure 2.24 The relationship between Artificial Intelligence, Machine Learning, Deep Learning, and Explainable Artificial Intelligence.

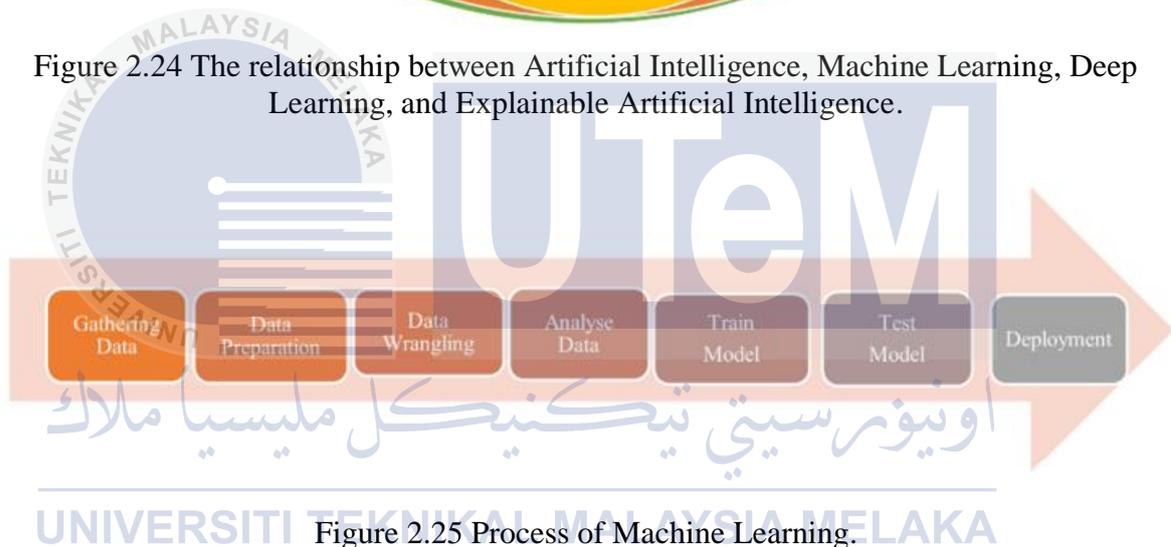


Figure 2.25 Process of Machine Learning.

2.7 You Only Look Once (YOLO)

The method of identifying objects YOLO has fundamentally altered computer vision. It is intended to conduct real-time object detection in pictures and videos, setting a new industry benchmark. Unlike traditional algorithms that rely on many stages, YOLO has a unified architecture that predicts bounding boxes and class probabilities simultaneously. Because of its unique detection technique, YOLO can reach amazing detection speed while maintaining high accuracy. Because of its effectiveness, simplicity, and ability to handle real-time situations, YOLO has acquired appeal and widespread implementation [44].

You Only Look Once (YOLO) is a popular and well-known algorithm. YOLO is well-known for its ability to detect objects. The first YOLO version was introduced in 2015 by Redmon et al. Scholars have produced various YOLO later versions referred to as YOLO

v2, YOLO v3, YOLO v4, YOLO v5, YOLO v6, YOLO v7 and YOLO v8. There are a few limited-edition revisions, such as YOLO-LITE [45]. Table 2.5 shows the evolution of YOLO object detection versions.

The model's small size and high calculation speed are at the heart of the YOLO target detection technique. YOLO's structure is easy. The neural network may immediately output the position and category of the bounding box. YOLO's speed is rapid since it simply needs to enter the image into the network to acquire the final detection result, therefore YOLO can also realize video time detection. YOLO detects objects directly using the global image, which may encode the global information and reduce the inaccuracy of perceiving the background as the object [46].

In this paper, the researcher focuses on YOLO v5 and YOLO v8. This is due to that the researcher will be using YOLO v8 for this project and will compare a previous work that uses YOLO v5. A brief introduction to YOLO v5 and YOLO v8, YOLO v5 was introduced in 2020. It was an unofficial version developed by Ultralytics, not endorsed by the original creators, with differences in architecture and implementation compared to YOLOv4 [45]. YOLO v8 was introduced in 2023. It was published by Ultralytics, considered the best YOLO model to date, incorporating new features and improvements for enhanced performance and flexibility. Suitable for various object detection, tracking, instance segmentation, image classification, and pose estimation tasks [44].

Table 2.5 The evolution of YOLO object detection versions.

YOLO Version	Year Introduced	Description
YOLOv1	2015	The first version of YOLO used a single neural network for object detection, providing fast but less accurate results than other systems.
YOLOv2	2016	Addressed limitations of YOLOv1 by incorporating a deeper neural network, batch normalization, and anchor boxes for improved accuracy.
YOLOv3	2018	Improved accuracy further by introducing a more complex neural network architecture, feature pyramid networks, and multi-scale predictions.
YOLOv4	2020	The latest version of YOLO introduced a more complex architecture, along with techniques like weighted feature fusion, spatial attention, and cross-stage partial connections for improved accuracy and speed.
YOLOv5	2020	An unofficial version developed by Ultralytics, not endorsed by the original creators, with differences in architecture and implementation compared to YOLOv4.
YOLOv6	2021	Another unofficial version based on YOLOv5, developed by researchers at the University of Massachusetts, introduces new techniques like cross-stage partial connections and hard negative example mining.
YOLOv7	2022	It was introduced by Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao and claimed to be the fastest and most accurate real-time object detector.
YOLOv8	2023	Published by Ultralytics, considered the best YOLO model to date, incorporating new features and improvements for enhanced performance and flexibility. Suitable for various object detection, tracking, instance segmentation, image classification, and pose estimation tasks.

2.7.1 YOLO v5 vs YOLO v8

As previously mention, based on the YOLO v5 algorithm, Jing Huang, Keyao Zeng, Zijun Zhang, and Wanhan Zhong create a solar panel fault detecting design. The YOLO v5 algorithm is improved in order to make up for the low detection efficiency of the traditional flaw detection techniques. The LCA attention mechanism is enhanced in three ways: first, by using coordinate attention to obtain a larger target range, which can enhance target feature sensing range in addition to fully capturing feature information; second, by giving different weights to the weighted bidirectional feature pyramid, which is more favourable to multi-scale Fast feature fusion; and third, by substituting the typical coupled head of the YOLO series with a decoupled head, which can enhance task branch and detection accuracy. ..Comparative analyses on the solar panel defect detection data set show that after algorithmic enhancements, the recall rate rises by 2.4% and 1.5% overall, and the mAP reaches up to 95.5%, 2.5% higher than pre-improvement levels. It can ensure electrical safety, standardize solar panel quality, and more accurately pinpoint any problems [47].

Table 2.6 compares the experimental results. Aside from the comparison with the original YOLO v5 model network, alternative target detection networks such as SSD, Faster-

RCNN, YOLO v3, YOLO v4, and YOLO v7 models were trained and tested using the identical solar panel defect detection dataset and model parameters. Except for Faster-RCNN, all of the models in the preceding networks are one-stage network models. Because the YOLO v5-BDL model has a larger network structure than the YOLO v5, its weight is 12.93 MB larger, and its detection speed is 26.73 fs 1, but other indicators such as Precision (P), Recall (R), and average accuracy (mAP) increase by 1.5%, 2.4%, and 2.5%, respectively. MAP increases by 3.3%, 22.32%, 5.3%, 3.1%, and 2.3%, respectively, when compared to SSD, Faster-RCNN, YOLOv3, YOLOv4, and YOLOv7 models, and the detection speed of a single image is faster and the file percentage is less. The comparison above clearly demonstrates that the YOLOv5-BDL model maintains superior performance in memory and other software costs, while boosting detection accuracy and performance [47].

YOLOv8 is the most recent model in the YOLO family, introduced by Ultralytics in 2022. YOLOv8 is based on the YOLOv5 framework and offers architectural and developer experience enhancements. It is faster and more accurate than YOLOv5, and it offers a unified framework for training models for object identification, instance segmentation, and image classification [47].

There are numerous models available for object detection. However, YOLOv8 and YOLOv5 are two of Ultralytics' most popular and cutting-edge versions. YOLOv8 is the newest member of the YOLO family, building on the success of previous versions while introducing new features and upgrades to improve performance and flexibility. In contrast, YOLOv5 is noted for its speed, simplicity, and precision. Table 2.7 shows the object detection performance comparison between YOLO v8 and YOLO v5 [47].

YOLOv8 and YOLOv5 are both fast object identification models that can process photos in real time. However, YOLOv8 is faster than YOLOv5, making it a superior alternative for real-time object detection applications. When selecting an object detection model, accuracy is a vital issue to consider. YOLOv8 is more accurate in this way than YOLOv5, thanks to various architectural advancements. Both YOLOv8 and YOLOv5 are simple to use, with YOLOv5 being the more user-friendly of the two. YOLOv5 is built on the PyTorch framework, making it simple to use and deploy for developers. YOLOv8

provides a unified training framework for models that perform object detection, instance segmentation, and picture classification. This makes it a better choice for those looking for a more extensive set of tools [47].

As a conclusion, when it comes to selecting the optimum object detection model, both YOLOv8 and YOLOv5 offer advantages and disadvantages. YOLOv5 is more user-friendly, however YOLOv8 is faster and more accurate. YOLOv8 is the best solution for applications that require real-time object detection. Finally, the model to utilize will be determined by the unique requirements of your application [47].

Table 2.6 Comparison of experimental result.

Model	P/%	R/%	mAP@0.5/%	FPS/(f.s ⁻¹)	Weights/MB
SSD	92.68	85.67	92.2	17.01	95.92
Faster-RCNN	66.83	71.46	73.18	7.27	111.01
YOLO v3	88.76	89.34	90.2	24.45	242.2
YOLO v4	94.55	85.52	92.4	23.57	250.39
YOLO v7	89.91	91.2	93.2	44.25	73.1
YOLO v5s	92.4	90.3	93	80.65	14
YOLO v5-BDL	93.9	92.7	95.5	53.92	26.93

Table 2.7 Object detection performance comparison between YOLO v8 and YOLO v5.

Model Size	YOLOv5	YOLOv8	Difference
Nano	28	37.3	+33.21%
Small	37.4	44.9	+20.05%
Medium	45.4	50.2	+10.57%
Large	49	52.9	+7.96%
Xtra Large	50.7	53.9	+6.31%

2.8 Thermal Imaging Sensor.

Since 1960, thermal imaging has been restricted to military and medical uses; but, with recent advances in chip technology and decreased costs, thermal imaging has gained mainstream popularity. Thermal imaging uses photons in the infrared section of the spectrum, specifically wavelengths ranging from 3 to 14 μm . Table 2.8 shows how these wavelengths are separated into distinct subbands. Thermal imagers use the infrared section of the spectrum to create a map of the spatial temperature distribution of the acquired scene. The temperature map's pixels represent the relative temperature of that spot in the environment. With proper calibration, bias reduction, and other processing, these temperature maps can be easily used for real-time applications [48].

Thermal imaging technique is independent of any external light source because it is based purely on the detection of infrared radiations (IRs) emitted by objects. As a result, the technology is proven to have a higher processing speed than its RGB counterparts. Due to decreasing chip costs, greater portability, and flexible designs, thermal imaging devices are now widely employed in civilian applications such as fever scanners, insulation detectors, and electrical hotspot detectors [48].

Thermal cameras' exceptional sensitivity has allowed them to be used in optical applications as well. Thermal imaging can also be used for fire prediction, weather forecasting, and animal monitoring. RGB cameras rely on illumination and reflection from objects, but thermal cameras detect emitted IR even when the object is cold. Thermal cameras have an edge over regular RGB cameras in discriminating between similar things since each object's heat signature is unique [48].

Table 2.8 Subbands in the infrared spectrum.

S No.	IR Band	Wavelength (in nm)
1	Near infrared	700 – 1400
2	Short range wavelength infrared	1400 – 3000
3	Mid range wavelength infrared	3000 – 8000
4	Long range wavelength infrared	8000 – 15000
5	Far infrared	15000 – 1000000

2.8.1 Principle of Thermal Sensing

Thermal imaging is a noncontact and nondestructive way of measuring an object's temperature. Thermal imaging uses the infrared radiation emitted by an object to create a visual temperature profile of the captured scene. The infrared spectrum is separated into distinct subbands based on wavelength, as indicated in Table 2.8. The intensity of IR emitted is determined by the wavelength [48].

Thermal imaging technology makes use of this energy intensity to produce a temperature map of the photographed image. The amount of thermal radiation emitted by a body is principally determined by its temperature (T) and emissivity factor (ϵ). The emissivity factor is the ratio of energy emitted by a body to that emitted by a perfect blackbody at the same temperature. For a perfect blackbody, the emissivity factor is one, and for a perfect white body, it is zero. Based on the IR energy radiated from a body, the surface temperature T_s of the body can be calculated as follows:

$$W = \left[\frac{2\pi^5 k^4}{15c^2 h^3} \right] T^4 = \sigma T_s^4 \quad (1)$$

where W represents the energy flux emitted per unit area (Wm^{-2}) of the body, c is the speed of light in vacuum ($3 \times 10^8 \text{ ms}^{-1}$), k is Boltzmann's constant ($1.38 \times 10^{-23} \text{ JK}^{-1}$), σ is Stefan-Boltzmann's constant ($5.67 \times 10^{-8} \text{ Wm}^{-2}\text{K}^{-4}$), h is Planck's constant ($6.63 \times 10^{-34} \text{ Js}$), and T is the temperature of the body in Kelvin [48]. When (1) is applied to real objects, then the surface temperature is computed as

$$W = \epsilon \sigma T^4 \quad (2)$$

where ϵ is the object's emissivity. By utilizing W , we can obtain a thermal visualization of the captured scene which is the basis of thermal imaging.

The thermal detector/sensor is the most important component of a thermal imaging system. The thermal detector is in charge of converting the incident IR to a temperature value. Thermal detectors are categorized into three types based on their operating principles: pyroelectric, thermoelectric, and bolometer sensors. Pyroelectric sensors are comprised of specific materials that build charge in response to incident infrared light. A change in temperature in the captured scene causes a proportionate change in the collected charge. This shift in the collected charge is utilized to compute the scene's thermal profile. The Seebeck

effect, on the other hand, governs the operation of thermoelectric sensors. The Seebeck effect is a phenomenon that produces a voltage difference based on the temperature difference between two dissimilar electrical conductors. In comparison to pyroelectric sensors, thermoelectric sensors are more reliable and less expensive. Nonlinearity difficulties plague thermoelectric sensors due to the nonlinear relationship between output voltage and sensed temperature [48].

Bolometer-based thermal detectors have recently acquired popularity due to their high thermal sensitivity, small size, and great accuracy. A bolometer is a particular material whose electrical resistance changes in response to the amount of IR incident on it. Vanadium oxide (VOx) and amorphous silicon (a-Si) are two common materials used in bolometers. The FLIR Lepton 3.5 is one example of a bolometer-based thermal sensor. For thermal imaging, the FLIR Lepton 3.5 employs a VOx-based microbolometer array. Figure 2.26 depicts a simplified block diagram of a microbolometer-based heat sensor in operation. The optical lens system focuses the incident infrared (IR) onto the focal plane array (FPA), as shown in Figure 2.26. Every element on the FPA is a pixel, and every pixel is a VOx microbolometer that changes temperature in response to incident flux. The microbolometer's resistance is proportionate to the temperature change. Voltage fluctuations pick up the change in resistance and feed it into a system-on-chip (SoC). The SoC processes the required signals and outputs the scene's thermal profile [48].

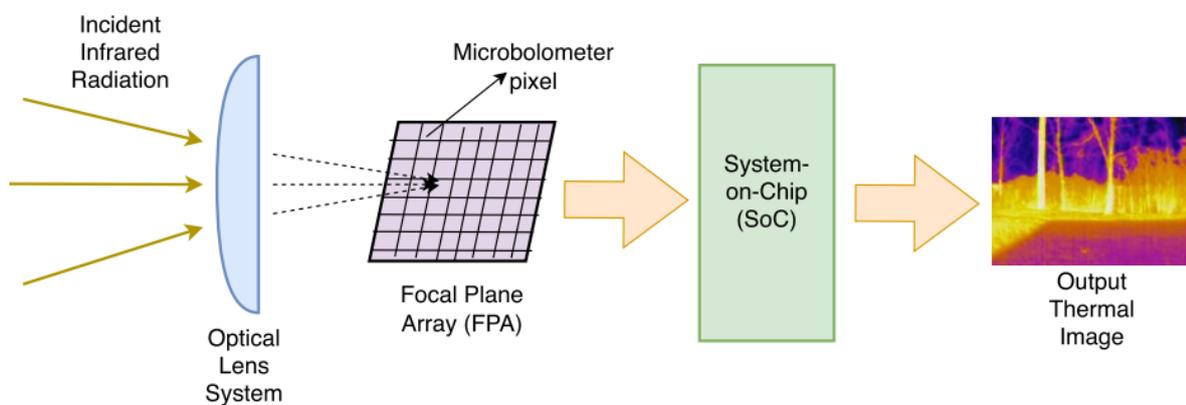


Figure 2.26 Working of the bolometer-based thermal sensor.

2.8.2 Latest Developments in Thermal Cameras

The variations of thermal camera models are displayed in Table 2.9 along with a few important metrics that can be used to determine which thermal camera is best for a certain application. The appropriateness of thermal camera models from FLIR and MOBOTIX for different applications has been covered in this paper [48].

Predictive maintenance makes use of FLIR thermal imaging cameras. In order to identify and address electrical problems, isolation problems, and other problems, they are also equally utilized by technicians and electricians. Accurate temperature profiles can be obtained from long-distance inspections using these cameras. Moreover, these cameras' multispectral dynamic imaging (MSX) function allows MSX to enhance the quality of the thermal images. Additionally, the interfaces are well-developed to provide simple output data transfer. The T-series, Ex, and Exx models of FLIR thermal cameras have this feature [48].

The E4, E5, E6, and E8 thermal cameras in the E-series are all very portable and useful for finding hidden flaws. This enables professionals to respond to a crisis before it gets too serious by acting quickly. These cameras are equipped with visual, thermal, and MSX imaging. The IR imaging resolution of the E4 (up to 4800 pixels), E5 (up to 10 800 pixels), E6 (up to 19 200 pixels), and E8 (up to 76 800 pixels) models can all be changed. The E40, E50, and E60 models are intended for wide-angle and frequent onsite technician and electrician inspection. These cameras also feature touchscreen controls and strong wifi connectivity, enabling immediate analysis of the thermal images they've taken [48].

The FLIR T-series is appropriate for measurements in harsh environments, including far-off places or warm temperatures. It can accurately aim the target for precise measurement and a superior perspective for analysis and capture thanks to its rotating optical block and autorotation feature. The integrated GPS in the T620 and T640 allows for better labelling of thermal images by adding a location. Applications requiring a fixed thermal camera mount can employ the FLIR A655sc. The FLIR A6200sc thermal camera is appropriate for InGaAs detection. High-speed mid-wave infrared (MWIR) technology is promising for the FLIR X8400sc series [48].

Thermal cameras from MOBOTIX are frequently utilized in surveillance settings. The M16 Thermal features two neighbouring lenses that perform thermal overlay on the optical image to identify hotspots, such as areas impacted by fire, in an image. A low-power camera with an extra thermal radiometry capability that makes it possible to measure the amount of thermal radiation in the image is the M16 TR thermal camera. Because of the sturdy housing surrounding the dual-camera sensor arrangement, the S16 DualFlex is a flexible dual-thermal camera with one or two waterproof sensors that can survive all environmental conditions[48].

As a long-term investment, selecting a thermal camera for a certain application necessitates careful consideration of a number of aspects. The proper supplier must be considered because the hardware of the thermal camera plays a major role in its operation. The various criteria for choosing a thermal camera are displayed in Figure 2.27 [48].

Table 2.9 Popular thermal cameras and their specifications.

Sl. No.	Model Name (Brand)	Camera Type	Size (mm)	Sensor Resolution (pixels)	Price (USD)	Detecting Temperature Range (°C)	Thermal Sensitivity (°C)	User Interface and Connectivity	References
1	FLIR C5 (Teledyne FLIR)	Compact Pocket Thermal Camera	138 x 84 x 24	160 x 120	855	-20 – 400	0 – 100 : ±3 ; 100 – 400 : ±3%	Touchscreen; FLIR Ignite cloud connectivity (using Wi-Fi)	[26]
2	Ti480 PRO (Fluke)	Hand-held Camera	277 x 122 x 167	640 x 480	–	-20 – 1000	±2 or 2%	Touch screen; Wireless connectivity (Smart Phone, PC); Fluke Connect® app compatible	[27]
3	CompactPRO XR (Seek Thermal)	Smart Phone Connected Thermal Camera	25.4 x 44.45 x 25.4	320 x 240	599	-40 – 330	< 0.070	Seek Thermal app	[28]
4	Helion 2 XP50 Pro (Pulsar)	Monocular Camera	242 x 75 x 60	640 x 480	4376	–	< 0.025	Built-in WiFi module - connects to Smart Phones using Stream Vision 2 app	[29]
5	CAT S62 PRO (CAT)	Thermal Imaging Smart Phone	158.5 x 76.7 x 11.9	1440 x 1080	530	-20 – 400	–	5.7" FHD + Display	[30]
6	Merger LRF XP50 (Pulsar)	Binocular Camera	196 x 143 x 76	640 x 480	6486	–	< 0.025	Built-in WiFi module - connects to Smart Phones using Stream Vision 2 app	[31]
7	RSE600 (Fluke)	Mountable Camera	83 x 83 x 165	640 x 480	–	-10 – 1200	±2 or ±2%	SmartView® desktop software	[32]
8	M16B Thermal TR (MOBOTIX)	Mountable Camera	210 x 158 x 207	336 x 252	–	-40 – 170	±0.05	HD wideband audio, Ethernet, RS232 support	[33]
9	S16B DualFlex (MOBOTIX)	Mountable Camera	130 x 115 x 33	336 x 252	–	-40 – 160	±0.05	HD wideband audio, Ethernet, RS232 support	[34]

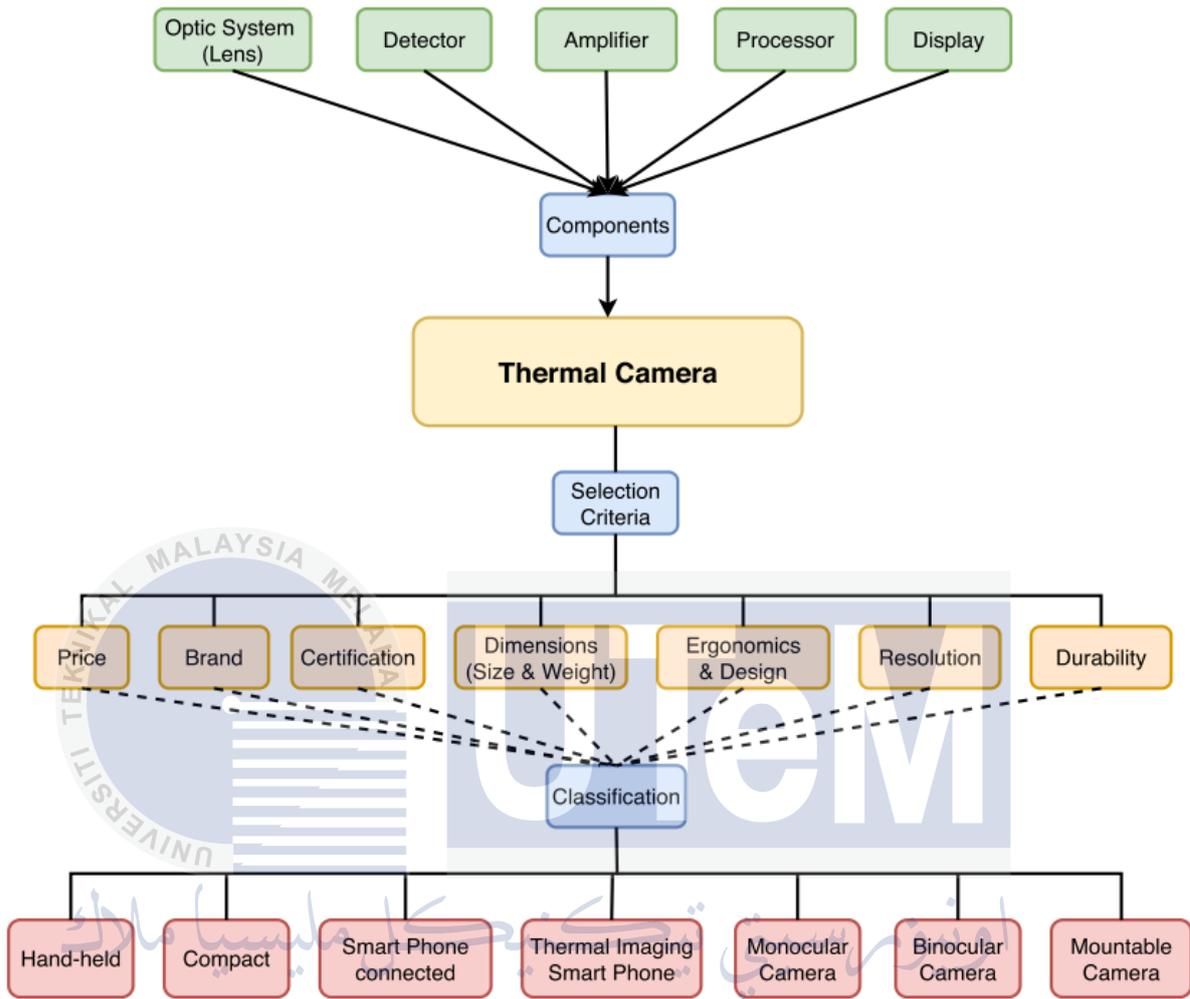


Figure 2.27 Components, selection criteria, and classification of thermal cameras.

2.9 Drone

Unmanned Aerial Vehicles (UAVs), or drones, were originally intended to be military weapons. Since then, they have been widely used for a variety of purposes, such as video surveillance, crowd management, public safety, and more. Drones have been incorporated into a number of distinct industries, including photography, videography, and transportation. By including additional functionalities such as camera visualization, microphone, and intelligent decision-making, these drones can achieve even greater heights in terms of surveillance and commerce [49].

Without a question, one of the sectors in the world that is expanding the fastest right now is the drone industry. Drone technology has limitless potential provided it continues to

expand and receive funding, which are necessary for it to be classified as an emerging technology. The best argument for an expanding commercial sector is thus made by the drone industry. Drone technology continues to serve an increasing number of sectors [50].

Using drones instead of traditional methods can result in significant cost savings and wider acceptance, all while boosting the value of the data acquired. This is due to the fact that traditional procedures can be time-consuming and prone to human mistake. As a result, a lot of new enterprises and startups are currently creating and discovering new applications and use cases for drones, propelling the drone industry ahead. Table 2.10 and 2.11 compares some of the drones that are available in the market.

Table 2.10 Drones that are available in the market.

	DJI Inspire 1	DJI Inspire 2	DJI Phantom 4 Pro	Typhoon H Pro
Flight Time »	18 mins	27 mins	30 mins	25 mins
Range »	2,000km	13,000m	13,000m	500m
Camera Quality »	Up to 16MP	20MP	20MP	12MP
Video Quality »	4K 30fps	5.2k 30fps	4k 60fps	4k 30fps

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 2.11 Drones that are available in the market.

	DJI Phantom 4	DJI Phantom 3 Pro	DJI Mavic Pro	Typhoon Q500 4K
Flight Time »	28 mins	24 mins	27 mins	25 mins
Range »	2,000m	2,000m	13,000m	500m
Camera Quality »	12MP	12MP	12MP	12MP
Video Quality »	4k 30fps	4k 30fps	4k 30fps	4k 30fps

2.10 Summary

This section is a summary for literature review. The use of renewable energy is becoming more and more important for promoting sustainable growth and reducing global warming. Renewable energy sources, like solar energy, are endless and environmentally benign in contrast to fossil fuels, which have a limited supply and harm the environment. The pressing need to cut greenhouse gas emissions and decrease reliance on non-renewable resources is what is driving the switch to renewable energy.

Utilizing photovoltaic (PV) panels and concentrating solar power systems, among other technologies, solar energy is one of the most promising renewable energy sources. Particularly, photovoltaic (PV) technology uses semiconductor materials to directly convert sunlight into electricity, making it a popular renewable energy source. Its increasing scalability and falling costs have added to its appeal on a worldwide scale.

PV systems have grown significantly on a global scale thanks to favourable regulations and advances in technology. According to available research, photovoltaic (PV) systems are becoming more and more cost-effective, making them a feasible choice for large-scale energy production. PV systems can range from massive utility-scale setups to residential and business installations, depending on the use.

PV panels can have a number of faults, though, which can reduce its lifespan and performance. Typical flaws include shade and shadowing, which lower energy production when impediments block sunlight, snail trails, which are discolouration lines brought on by moisture and chemical reactions, and hotspots, which are localised regions of high temperature brought on by shading or cell destruction. Other defects such as microcracks, small fractures in the cells that decrease efficiency; delamination, where layers in the panel separate often due to moisture ingress; potential induced degradation (PID), which involves voltage stress causing performance loss; and diode failure, where malfunctioning bypass diodes lead to energy loss, also affect PV panels.

Finding these flaws is essential to keeping PV systems operating efficiently. Traditional inspection techniques have several drawbacks since they are labour- and time-intensive. Recent developments have made it possible to automate the fault identification

process by using thermal imaging and artificial intelligence (AI). Artificial intelligence models, namely those that utilise the You Only Look Once (YOLO) algorithm, have proven to be highly accurate in detecting and categorising flaws in photovoltaic panels.

The accuracy and efficiency of the fault identification procedure are greatly improved by AI. The YOLO method is well known for its real-time object identification capabilities, especially in its most recent versions (YOLO v5 and v8). YOLO v8 is a better option for defect detection tasks because to its significant speed and accuracy improvements over v5, as demonstrated by comparisons between the two versions.

Thermal imaging sensors are essential for detecting temperature changes on photovoltaic panels. Because these sensors pick up on infrared radiation, they are crucial for identifying faults such as hotspots and other thermal irregularities. The resolution and sensitivity of thermal cameras have been enhanced recently, which increases their usefulness for PV panel examinations.

Thermal camera-equipped drones provide an economical and effective way to examine massive solar arrays. These drones are able to take thermal pictures on their own, which are then processed by AI systems to find flaws. This creative concept offers a workable answer for the solar energy sector by drastically cutting the time and labour involved with conventional inspection techniques.

In conclusion, there has been a significant development in the detection of PV panel defects thanks to the integration of AI with modern imaging technologies like heat sensors and drones. These innovations enhance the overall performance and dependability of solar energy systems in addition to increasing the precision and effectiveness of inspections. The optimisation and further growth of solar energy as a major renewable energy source depend on this integration.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter explains the method that was used to develop the model for defect detection. The model will be developed by using YOLO v8 to detect defect using thermal imaging sensor and will be attached to a drone.

3.2 Flowchart

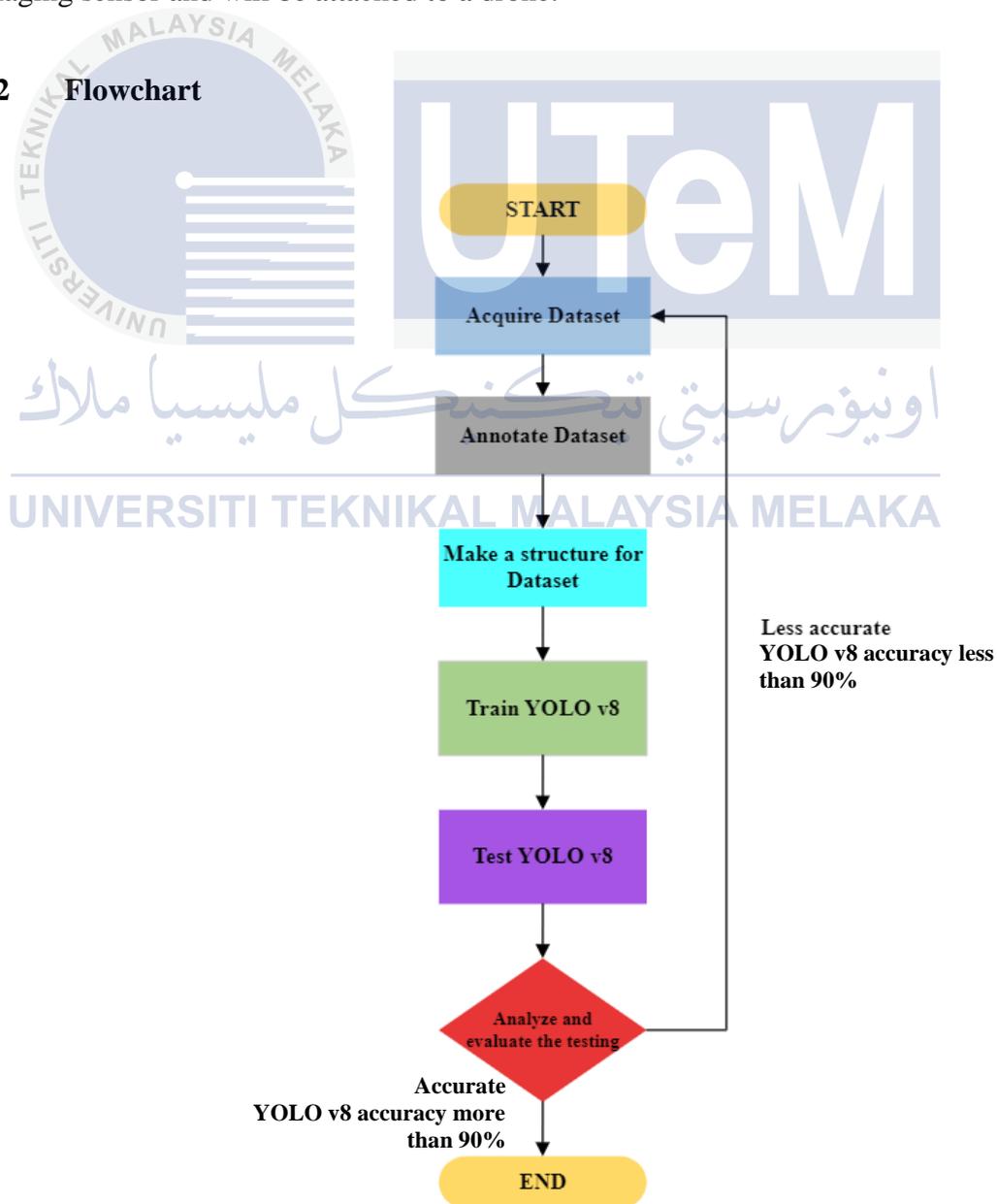


Figure 3.1 Flowchart for model generation.

3.3 Acquiring Dataset

For FYP 1, the researcher will acquire the dataset by downloading images of defected photovoltaic panels from the Internet. The images vary in size and environmental condition. A total of 339 images were acquired.

In FYP 2, the researcher has visited Gading Kencana solar farm located in Ayer Keroh, Malacca. The solar farm has a total of 29,092 photovoltaic panels which supplies 8MW of electricity. The solar spans 14.1 acres and is designed to maximize land use efficiency through innovative panel arrangement and mounting techniques. The researcher has also brought a drone for capturing thermal images. By using Drone Harmony for route planning, the drone which is a DJI Mavic 2 Enterprise model is flown autonomously above the photovoltaic panel. Drone Harmony is a sophisticated 3D flight management software for drones and UAVs that is especially beneficial for mapping, inspection, and surveying operations. It was founded in Switzerland in 2016 and provides innovative tools for automated flight planning, data collecting, and data evaluation, making it an adaptable solution for a wide range of industries, including mining, construction, industrial inspection, and more. Some of the key features of Drone Harmony is full 3D flight management interface and automated mission planning. The researcher managed to acquire a total of 3064 images of defected photovoltaic panels. It is also important to note that the researcher capture the thermal images on a bright sunny day with a luminescent of 20000 to 60000 lux. Capturing the thermal images with a luminescent of more than 60000 lux will cause the solar panels to appear too hot under thermal imaging thus the real hotspot not be able to detect.

This is a very significant increase in number of datasets which is nearly,10 times more than the dataset acquired in FYP 1. This number of datasets benefits the researcher in providing a better a result than FYP 1. Figure 3.2 shows the route planning using Drone Harmony. Figure 3.3 shows the DJI Mavic 2 Enterprise taking off autonomously by using Drone Harmony. Figure 3.4 shows the DJI Mavic 2 Enterprise flying above the photovoltaic panels autonomously by using Drone Harmony. Figure 3.5 shows the lux meter with a reading of 55700 lux during the drone capturing thermal images.

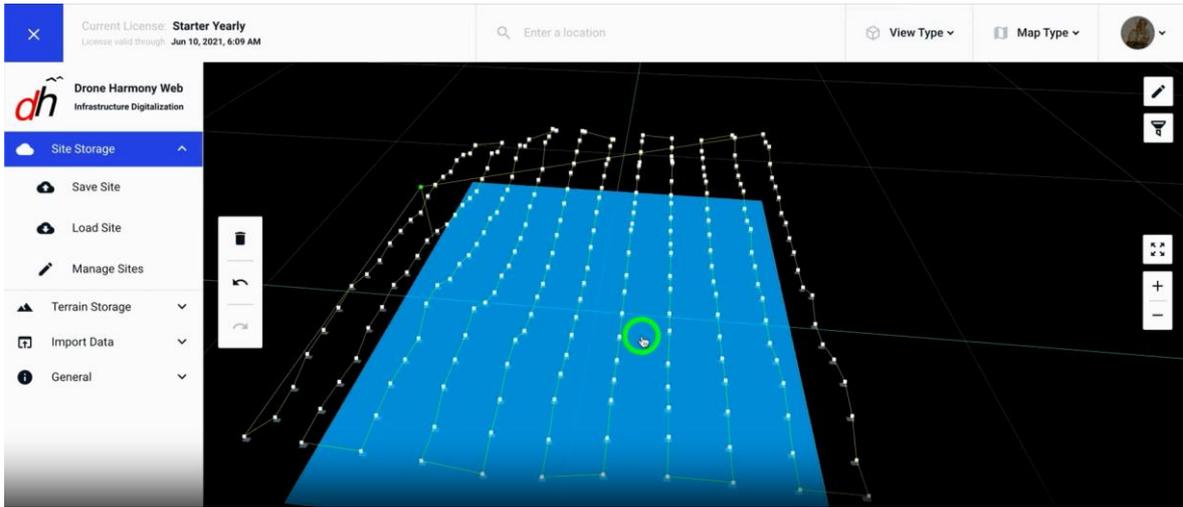


Figure 3.2 Drone Harmony route planning



Figure 3.3 Drone take off



Figure 3.4 Drone flying autonomously above the photovoltaic panel



Figure 3.5 Lux meter with 55700 lux

3.4 Annotating Dataset

Annotating is a process where a specific object from an image is highlighted by using an annotating tool. The researcher will be using Roboflow to annotate the acquired dataset. Roboflow enabled the researcher to efficiently edit and manage the dataset, with AI support for bounding boxes, polygons, and instance segmentation. Roboflow supports a variety of input formats, including JPG, PNG, BMP, MOV, MP4, and AVI, making it simple to import the photos required for dataset construction. The software also supports easy export in a variety of formats, including coco JSON, VGG, Vott JSON, Marmot XML, YOLO PyTorch, YOLO Darknet TXT, and Kaggle CSV, making it easy to use the data in a variety of machine

learning frameworks. One of the most significant benefits of utilizing Roboflow was its simple UI and clever defaults. This enabled the researcher to annotate photographs swiftly and properly without the requirement for substantial training or specialized knowledge. Furthermore, Roboflow has strict privacy and security standards in place, guaranteeing that our data is safe and secure.

To achieve the best results possible, the researcher manually segmented each image from edge to edge, ensuring that the YOLO model had accurate and thorough data to work with. This attention to detail and emphasis on accuracy will ensure that the algorithm can detect defect on photovoltaic panels in real-world circumstances. With the help of Roboflow's sophisticated features and careful manual image segmentation, the researcher was able to produce a high-caliber dataset that will allow our model to function at its peak even in demanding situations.

In FYP 1, the researcher annotated 339 images of defected solar panels. This process took 5 hours to complete. In FYP 2, the researcher annotates 3064 images of defected solar panels. As expected, this process took significantly more time to complete. It took the researcher 4 days to complete the annotation for all 3064 images. This annotation process will help the AI model to learn and recognize hotspots. Figure 3.5, 3.6 and 3.7 shows the Roboflow tool. Figure 3.8 and 3.9 shows the researcher annotating the hotspots.

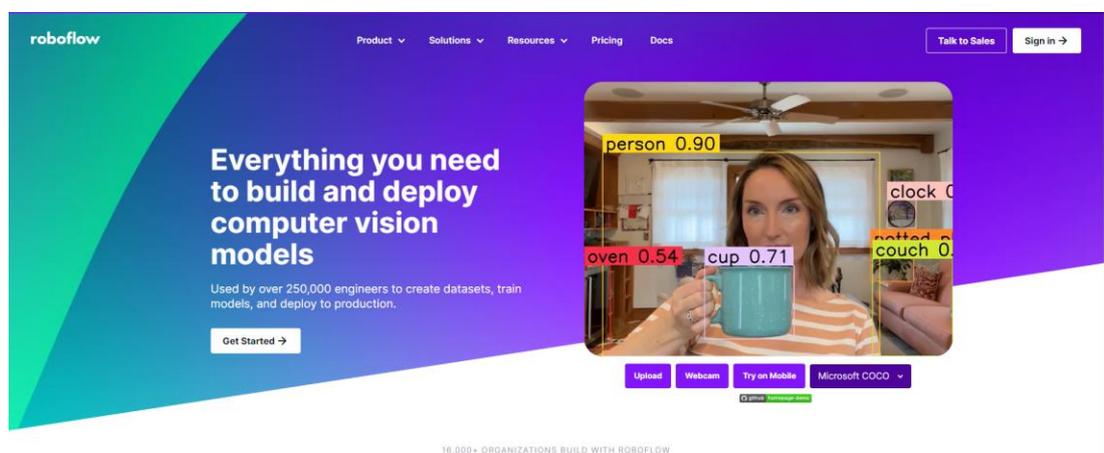


Figure 3.6 Roboflow tool.

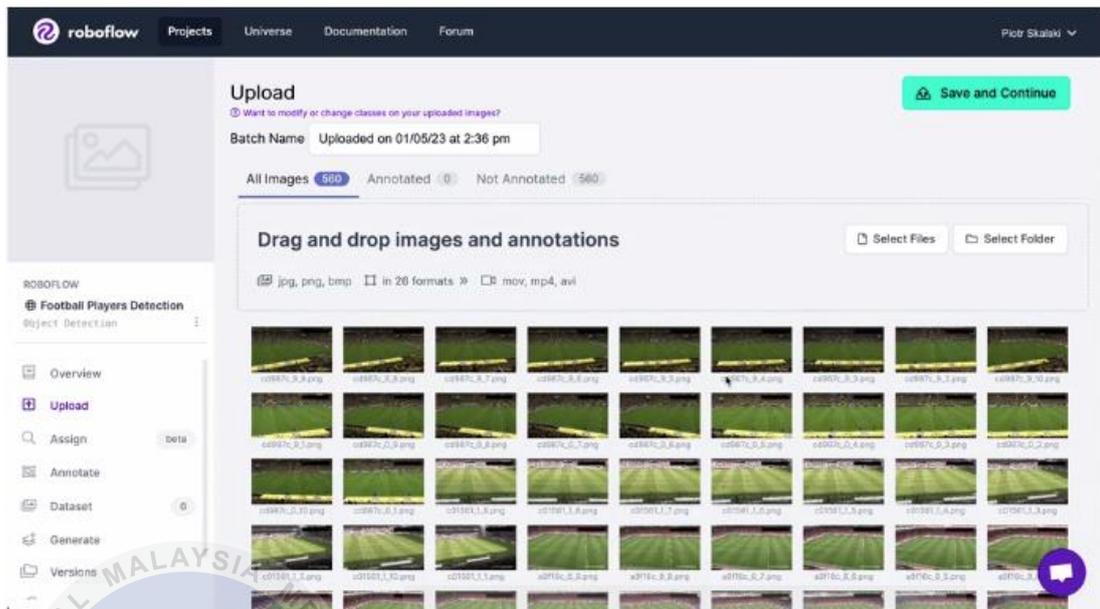


Figure 3.7 Roboflow tool (Uploading images).



Figure 3.8 Roboflow tool (Annotating images).

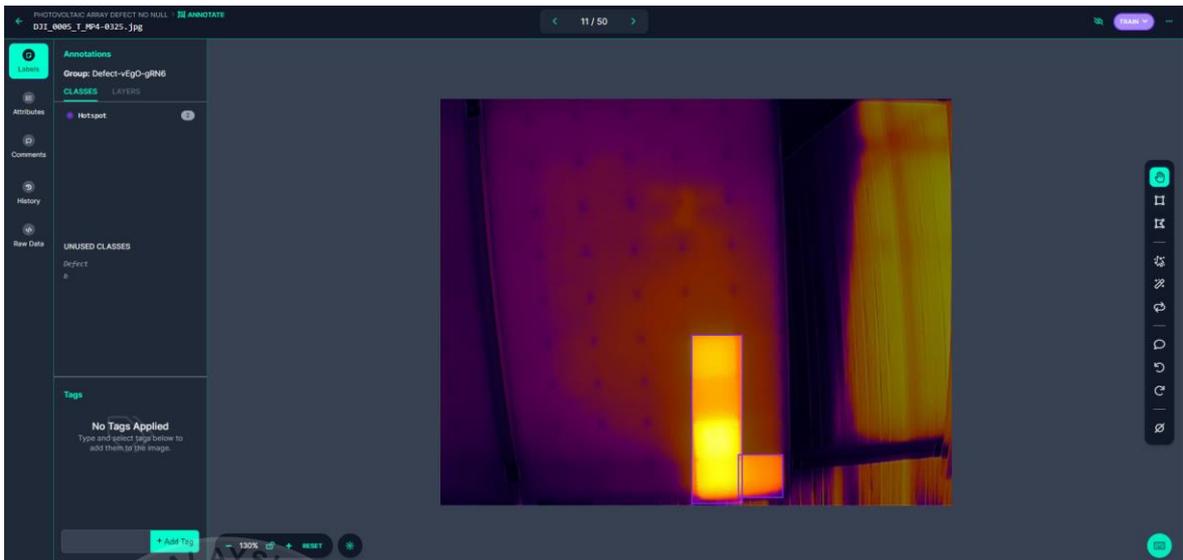


Figure 3.9 Annotation of hotspots

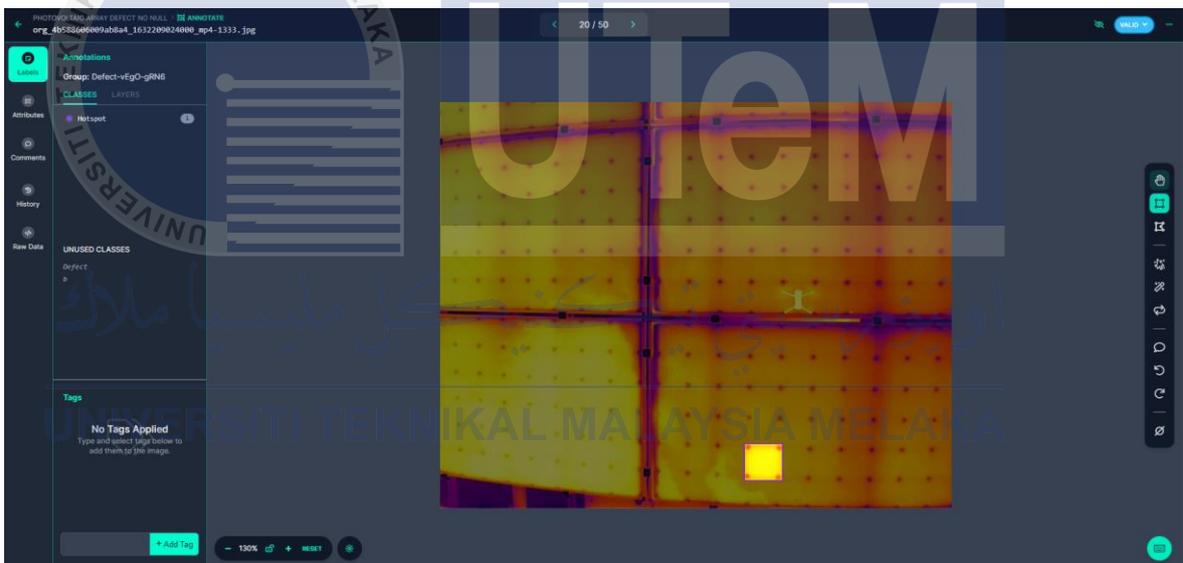


Figure 3.10 Annotation of hotspots

3.5 Structure of Dataset

The researcher's model which uses YOLO v8 will undergo intensive training and testing. Before the training process begin, the researcher will need to make a structure for the dataset. There are two types of dataset structure that can be used to train an AI model. The first structure is train and test the model using the same dataset. The researcher will transfer all the dataset into the model and the model will begin training. Once the done, the same dataset will be used to train the model.

The second structure is called Train/Test split. The Train/Test split structure is when the model learned only a portion of the dataset and the rest of the dataset will be used for testing. It is said that the Train/Test split structure allows the model to give a more accurate result and prediction in real life circumstances. This is due to the model able to test itself on a dataset that the model has not learned which is crucial in providing accurate result or prediction. The researcher has decided to use the Train/Test split structure for training and testing the model.

In both FYP 1 and FYP 2, the researcher split the dataset into 70% for training, 20% for validation and 10% for testing. This structure allows the AI model to learn and train many datasets while allowing to test itself using datasets that the AI model has not learned. Figure 3.10 shows the structure of dataset for FYP 2 using Roboflow. Table 3.1 shows the structure of dataset for both FYP. Figure 3.11 shows the code to train the model.

Table 3.1 Structure of dataset for both FYP

FYP	No. Of Dataset	Train (70%)	Validation (20%)	Testing (10%)
1	339	237	68	34
2	3064	2151	603	310

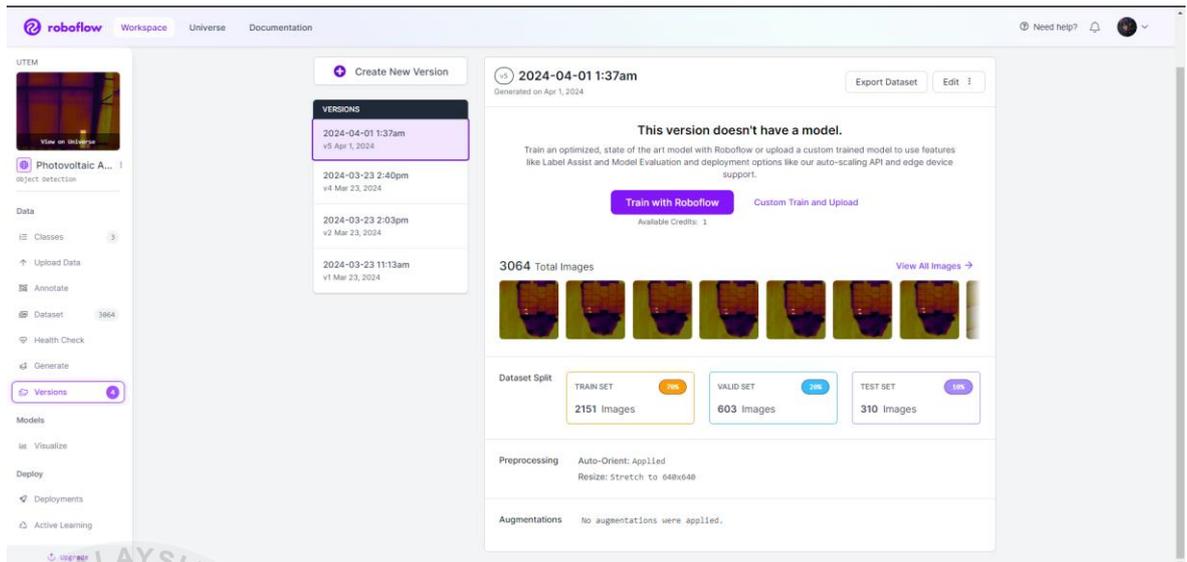


Figure 3.11 Structure of dataset for FYP 2

3.6 Training and Testing

3.6.1 Google Collab

YOLO v8 uses Python language. In FYP 1, the researcher uses Google Collab to train the AI model. Google Collaboratory, or Collab, is an as-a-service version of Jupyter Notebook that enables you to write and execute Python code through your browser. Google Collab is built around Project Jupyter code and hosts Jupyter notebooks without requiring any local software installation. But while Jupyter notebooks support multiple languages, including Python, Julia and R, Collab currently only supports Python. Google Collab is free, but has limitations. There are some code types that are forbidden, such as media serving and crypto mining. Available resources are also limited and vary depending on demand, though Google Collab offers a pro version with more reliable resourcing. The researcher trains the YOLO v8 model at 4 different epochs, 25, 50, 75 and 100 epochs with 339 datasets. The Google Collab has already all the necessary code to train and test the YOLO v8 model. Therefore, the researcher will only require to copy the api key for the datasets from Roboflow and paste it into the code. Once done, the researcher will begin the training and testing process. Figure 3.11 shows the training and testing using Goolge Collab. Figure 3.12 shows the code for installing and importing Roboflow using Google Collab. Figure 3.13 shows the code to train the model using Google Collab.

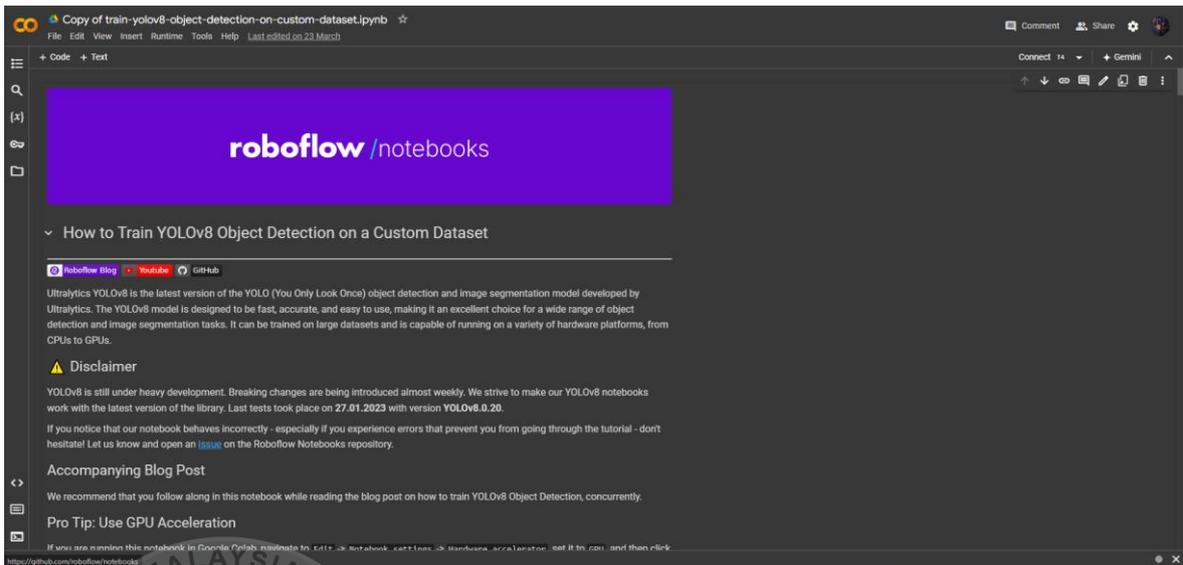


Figure 3.12 Training and testing using Goolge Collab

```
[ ] !mkdir {HOME}/datasets
    %cd {HOME}/datasets
    !pip install roboflow

from roboflow import Roboflow
rf = Roboflow(api_key="fQ12Cn7bBC3yrEuN520e")
project = rf.workspace("utem-8dxoz").project("photovoltaic-array-defect")
dataset = project.version(1).download("yolov8")
```

Figure 3.13 Installing and importing Roboflow

```
!pip install roboflow
from roboflow import Roboflow
rf =
Roboflow(api_key="fQ12Cn7bBC3yrEuN520e")
project = rf.workspace("utem-
8dxoz").project("photovoltaic-array-
defect")
dataset =
project.version(5).download("yolov8")
```

Figure 3.14 Code to train the model using Google Collab

3.6.2 Offline

In FYP 2, the researcher trains the AI model via offline. The researcher uses command prompt, a built-in command line interpreter application available in most Windows operating systems. The researcher trains the YOLO v8 model at the same 4 different epochs, 25, 50, 75 and 100 epochs with 3064 datasets. In order to begin training and testing the YOLO v8 model using command prompt, the researcher must first set the desired directory. Once set, the researcher then has to create a python environment and activate the necessary python scripts. Then, the researcher needs to install the latest version of python by using “pip install –upgrade” command. After that, the researcher will need to install Ultralytics. Once installed, the researcher can begin training and testing the AI model.

Initially, the researcher trains and tests the AI model using CPU but due limitation, the researcher opted to train and test the AI model using GPU. In order for the researcher to convert the training and testing process from using CPU to GPU, the researcher needs install Torch, TorchVision and TorchAudio. Once installed, the researcher runs a code to check if the GPU is available. If it is available, the researcher begins training and testing the AI model. It is important to note that the researcher must run the command prompt as administrator to avoid any issues the training and testing. Figure 3.14 shows the command prompt to train offline. Figure 3.15 shows the necessary steps to train and test using CPU. Figure 3.16 the researcher activated the Python environment and ready to begin training and testing. Figure 3.17 shows the steps to convert CPU to GPU. Figure 3.18 shows that GPU is available for training and testing. Figure 3.19 shows the code for training and testing the YOLO v8 offline.

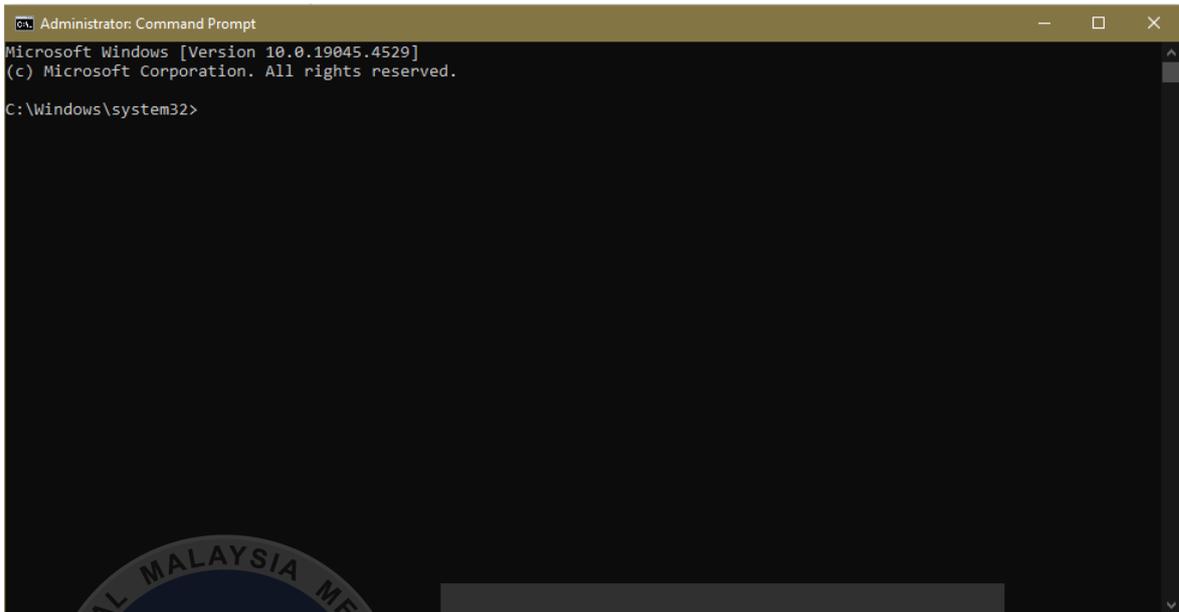


Figure 3.15 Command prompt to train offline

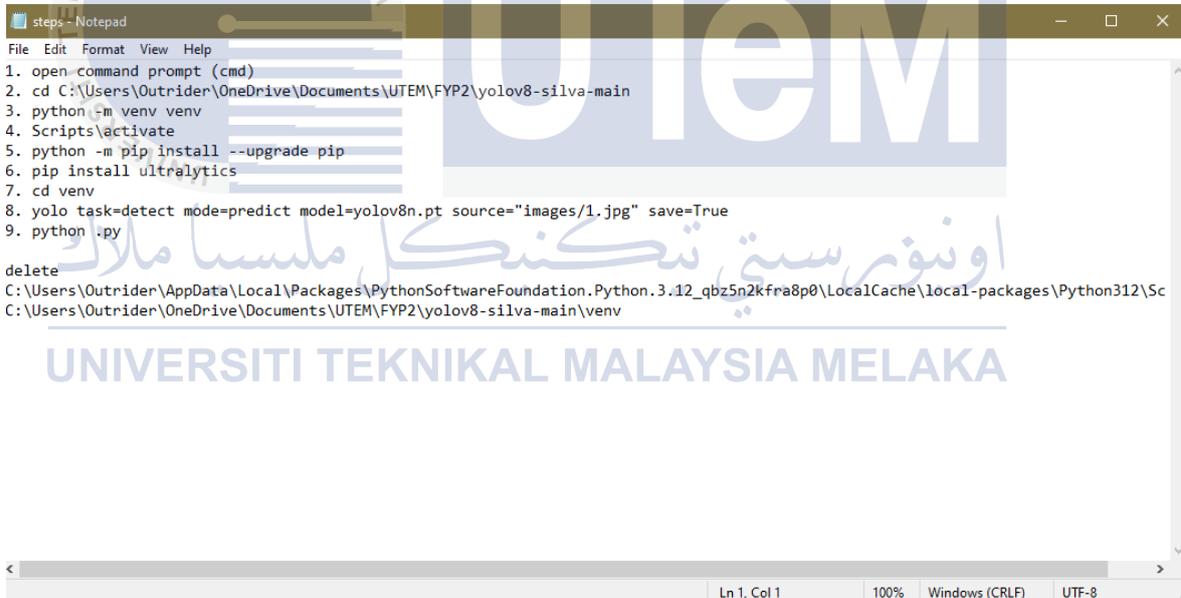
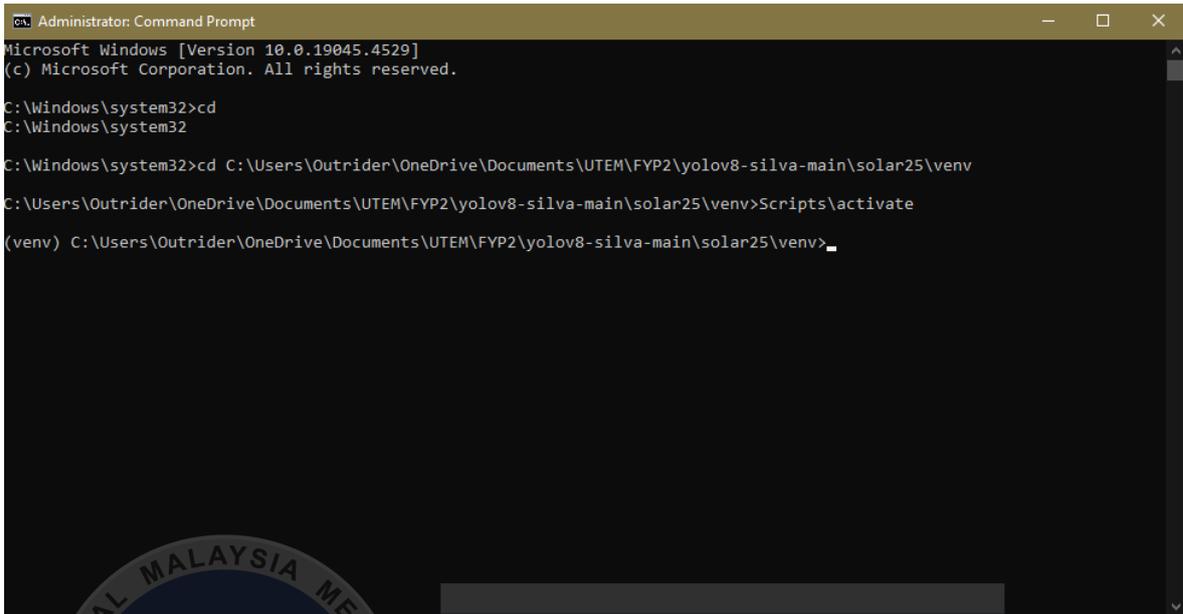


Figure 3.16 Steps to train and test using CPU

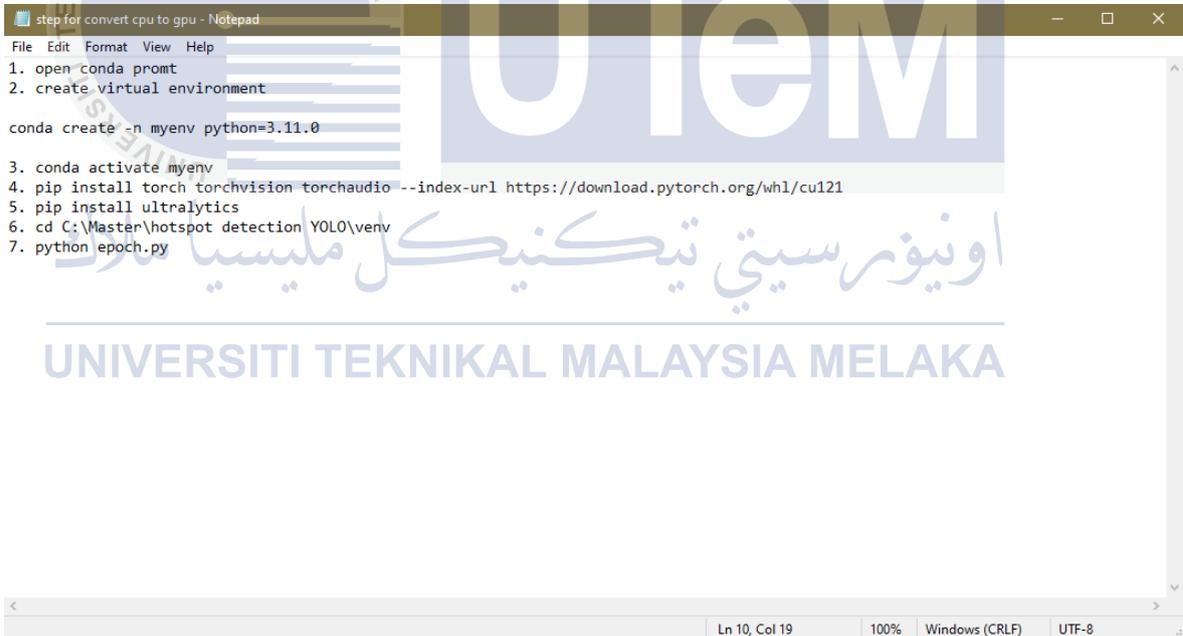


```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.19045.4529]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd
C:\Windows\system32

C:\Windows\system32>cd C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv
C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv>Scripts\activate
(venv) C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv>_
```

Figure 3.17 Python environment activated



```
step for convert cpu to gpu - Notepad
File Edit Format View Help
1. open conda prompt
2. create virtual environment

conda create -n myenv python=3.11.0

3. conda activate myenv
4. pip install torch torchvision torchaudio --index-url https://download.pytorch.org/whl/cu121
5. pip install ultralytics
6. cd C:\Master\hotspot detection YOLO\venv
7. python epoch.py
```

Figure 3.18 Steps to convert CPU to GPU

```

Administrator: Command Prompt
Microsoft Windows [Version 10.0.19045.4529]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd
C:\Windows\system32

C:\Windows\system32>cd C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv
C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv>Scripts\activate

(venv) C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv>python class1.py
GPU is available
Device name: NVIDIA GeForce GTX 1650
Memory allocated: 0
C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv\Lib\site-packages\torch\cuda\memory.py:440
: FutureWarning: torch.cuda.memory_cached has been renamed to torch.cuda.memory_reserved
  warnings.warn(
Memory cached: 0
Using device: cuda

(venv) C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv>

```

Figure 3.19 GPU is available

```

solar25 - Notepad
File Edit Format View Help
from ultralytics import YOLO

if __name__ == '__main__':
    model=YOLO()
    model.train(data="C:\\Users\\Outrider\\OneDrive\\Documents\\UTEM\\FYP2\\yolov8-silva-main\\solar25\\venv\\pv\\data.yaml", epochs=5, save=True, device=0)

```

Figure 3.20 Code to train the model using Offline

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.7 Summary

During the first phase of the Final Year Project (FYP 1), the researcher generated a dataset by downloading 339 photos of defective solar panels from the Internet. These photos ranged in size and environmental conditions, giving a diverse foundation for preliminary research. During the second phase (FYP 2), the researcher went to the Gading Kencana solar farm in Ayer Keroh, Malacca, which covers 14.1 acres and has 29,092 photovoltaic panels capable of producing 8MW of electricity. The farm's design maximizes land usage efficiency by utilizing unique panel arrangement and mounting techniques.

The study used a DJI Mavic 2 Enterprise drone with Drone Harmony, a sophisticated 3D flight control software, to acquire thermal photos of the photovoltaic panels. Drone Harmony, created in Switzerland in 2016, offers advanced capabilities for automated flight planning, data gathering, and evaluation, making it ideal for mapping, inspection, and

surveying jobs in a variety of industries. Using this approach, the researcher collected 3,064 photos of defective panels, a considerable increase over the collection from FYP 1.

The study used Roboflow for the annotating process, which is an efficient tool that supports a variety of input formats and gives AI aid with bounding boxes, polygons, and instance segmentation. Roboflow's user-friendly interface and strict privacy requirements made it easier to annotate photographs quickly and accurately, which is required for training the YOLO model. The researcher methodically divided each image to provide complete data for the YOLO model, which is critical for reliable defect identification in real-world scenarios. This technique required five hours for 339 photographs in FYP 1 and four days for 3064 images in FYP 2.

The researcher adopted the Train/Test split structure for dataset organization, dividing the dataset into 70% for training, 20% for validation, and 10% for testing. This strategy allows the model to learn from a major amount of the data while validating its accuracy on unseen data, boosting its real-world prediction skills.

In FYP 1 and FYP 2, training the YOLO v8 model required the use of Python in several contexts. Using Google Collab, a cloud-based Jupyter Notebook service, the researcher trained the model in FYP 1 across 25, 50, 75, and 100 epochs using the initial dataset. Although useful, Google Collab has restrictions on the types of code that may be executed and the availability of resources.

For the larger dataset in FYP 2, the researcher switched to offline training with a GPU and a command prompt. Installing relevant libraries, configuring a Python environment, and verifying GPU compatibility were all necessary for this. The investigator employed a methodical approach, which involved setting up Torch, TorchVision, and TorchAudio, to effectively utilize GPU capabilities for YOLO v8 model training and testing. The goal of this painstaking configuration was to improve the model's performance by getting around the drawbacks of CPU-based training.

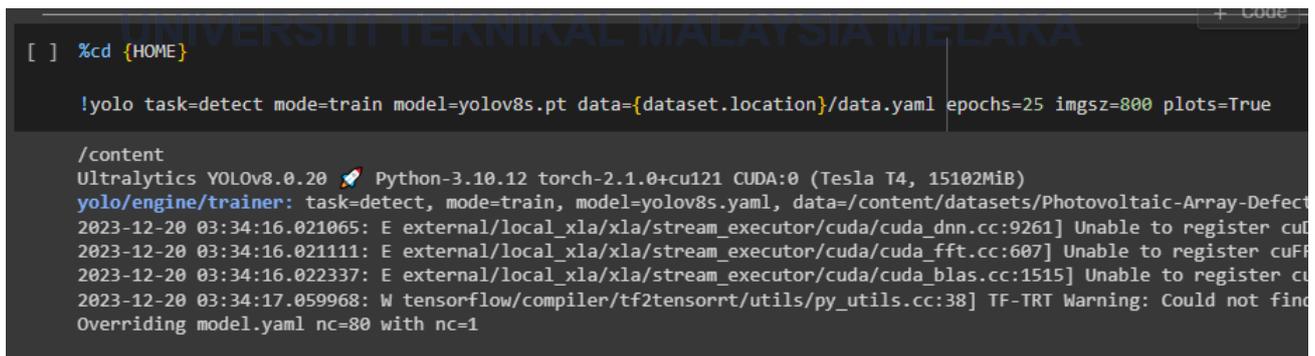
CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Custom Training Dataset

4.1.1 Google Collab

As mentioned in Chapter 3, in FYP 1 the researcher used Google Collab to train and test the YOLO v8 model. The researcher trains and tests at 4 different epochs, 25, 50, 75 and 100 epochs with 339 datasets. The researcher was able to achieve all results for all epochs. All results were automatically generated and saved in Google Collab. The researcher also has the option to save in PC. However, when the researcher wants to the train the AI model with 3064 datasets at the same epochs, there was a limitation. The usage time exceeds the time limit. Therefore, in FYP 2 the researcher trains the YOLO v8 model via offline. Figure 4.1 to Figure 4.5 shows the training and testing at 25 epochs with 339 datasets process in Google Collab.



```
[ ] %cd {HOME}
!yolo task=detect mode=train model=yolov8s.pt data={dataset.location}/data.yaml epochs=25 imgsz=800 plots=True

/content
Ultralytics YOLOv8.0.20 Python-3.10.12 torch-2.1.0+cu121 CUDA:0 (Tesla T4, 15102MiB)
yolo/engine/trainer: task=detect, mode=train, model=yolov8s.yaml, data=/content/datasets/Photovoltaic-Array-Defect
2023-12-20 03:34:16.021065: E external/local_xla/xla/stream_executor/cuda/cuda_dnn.cc:9261] Unable to register cu
2023-12-20 03:34:16.021111: E external/local_xla/xla/stream_executor/cuda/cuda_fft.cc:607] Unable to register cuFF
2023-12-20 03:34:16.022337: E external/local_xla/xla/stream_executor/cuda/cuda_blas.cc:1515] Unable to register cu
2023-12-20 03:34:17.059968: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not find
Overriding model.yaml nc=80 with nc=1
```

Figure 4.1 Training and testing process in Google Collab

Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
1/25	8.55G	2.045	3.412	1.73	10	800: 100% 45/45 [00:41<00:00, 1.09it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.50it/s]
	all	68	161	0.503	0.429	0.351 0.134
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
2/25	8.55G	1.888	1.903	1.527	48	800: 100% 45/45 [00:36<00:00, 1.23it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.75it/s]
	all	68	161	0.477	0.46	0.343 0.161
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
3/25	8.55G	1.87	1.864	1.51	38	800: 100% 45/45 [00:37<00:00, 1.20it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.73it/s]
	all	68	161	0.547	0.195	0.25 0.119
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
4/25	9.73G	1.843	1.953	1.453	8	800: 100% 45/45 [00:38<00:00, 1.18it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.60it/s]
	all	68	161	0.522	0.447	0.442 0.214
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
5/25	9.73G	1.81	1.6	1.435	14	800: 100% 45/45 [00:39<00:00, 1.15it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.73it/s]
	all	68	161	0.453	0.503	0.438 0.193
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
6/25	9.73G	1.772	1.528	1.387	26	800: 100% 45/45 [00:35<00:00, 1.27it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:02<00:00, 1.27it/s]
	all	68	161	0.652	0.499	0.529 0.245

Figure 4.2 Training and testing process in Google Collab

Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
7/25	9.73G	1.749	1.46	1.389	11	800: 100% 45/45 [00:35<00:00, 1.25it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.72it/s]
	all	68	161	0.555	0.581	0.546 0.261
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
8/25	9.73G	1.731	1.391	1.394	28	800: 100% 45/45 [00:36<00:00, 1.24it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.74it/s]
	all	68	161	0.562	0.423	0.469 0.207
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
9/25	9.73G	1.692	1.345	1.353	22	800: 100% 45/45 [00:37<00:00, 1.20it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.75it/s]
	all	68	161	0.619	0.646	0.61 0.287
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
10/25	9.73G	1.69	1.294	1.365	27	800: 100% 45/45 [00:38<00:00, 1.18it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.72it/s]
	all	68	161	0.659	0.559	0.551 0.271
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
11/25	9.73G	1.696	1.235	1.356	19	800: 100% 45/45 [00:35<00:00, 1.25it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:02<00:00, 1.32it/s]
	all	68	161	0.699	0.553	0.585 0.295
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
12/25	9.73G	1.615	1.165	1.305	11	800: 100% 45/45 [00:35<00:00, 1.25it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.62it/s]
	all	68	161	0.684	0.592	0.626 0.291
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
13/25	9.73G	1.629	1.183	1.34	11	800: 100% 45/45 [00:36<00:00, 1.23it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.75it/s]
	all	68	161	0.674	0.578	0.643 0.309
Epoch	GPU_mem	box_loss	cls_loss	df1_loss	Instances	Size
14/25	9.73G	1.629	1.119	1.303	25	800: 100% 45/45 [00:37<00:00, 1.21it/s]
	Class	Images	Instances	Box(P	R	mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.66it/s]
	all	68	161	0.703	0.621	0.636 0.299

Figure 4.3 Training and testing process in Google Collab

```

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
15/25 9.73G 1.555 1.04 1.276 20 800: 100% 45/45 [00:37<00:00, 1.10it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.73it/s]
all 68 161 0.719 0.596 0.65 0.33
Closing data_loader mosaic
albumations: Blur(p=0.01, blur_limit=(3, 7)), MedianBlur(p=0.01, blur_limit=(3, 7)), ToGrayscale(p=0.01, CLAHE(p=0.01, clip_limit=(1, 4.0), tile_grid_size=(8, 8))

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
16/25 9.73G 1.567 1.032 1.319 7 800: 100% 45/45 [00:27<00:00, 1.66it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.73it/s]
all 68 161 0.598 0.578 0.556 0.265

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
17/25 9.73G 1.57 0.994 1.314 17 800: 100% 45/45 [00:26<00:00, 1.71it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.73it/s]
all 68 161 0.695 0.581 0.599 0.314

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
18/25 9.73G 1.562 0.9647 1.293 17 800: 100% 45/45 [00:24<00:00, 1.87it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.72it/s]
all 68 161 0.707 0.602 0.624 0.327

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
19/25 9.73G 1.489 0.9137 1.268 5 800: 100% 45/45 [00:24<00:00, 1.86it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:01<00:00, 1.67it/s]
all 68 161 0.735 0.621 0.624 0.318

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
20/25 9.73G 1.452 0.8568 1.246 11 800: 100% 45/45 [00:23<00:00, 1.90it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:02<00:00, 1.47it/s]
all 68 161 0.765 0.648 0.666 0.342

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
21/25 9.73G 1.443 0.8446 1.231 14 800: 100% 45/45 [00:23<00:00, 1.93it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:02<00:00, 1.32it/s]
all 68 161 0.789 0.627 0.67 0.353

```

Figure 4.4 Training and testing process in Google Collab

```

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
23/25 9.73G 1.4 0.7832 1.205 5 800: 100% 45/45 [00:22<00:00, 1.98it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:02<00:00, 1.43it/s]
all 68 161 0.799 0.616 0.68 0.357

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
24/25 9.73G 1.381 0.7572 1.191 7 800: 100% 45/45 [00:24<00:00, 1.84it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:02<00:00, 1.37it/s]
all 68 161 0.817 0.609 0.669 0.351

Epoch GPU_mem box_loss cls_loss dfl_loss Instances Size
25/25 9.73G 1.356 0.7353 1.169 12 800: 100% 45/45 [00:22<00:00, 1.97it/s]
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:03<00:00, 1.13s/it]
all 68 161 0.772 0.63 0.686 0.379

25 epochs completed in 0.244 hours.
Optimizer stripped from runs/detect/train2/weights/last.pt, 22.5MB
Optimizer stripped from runs/detect/train2/weights/best.pt, 22.5MB

Validating runs/detect/train2/weights/best.pt...
Ultralytics YOLOv8.0.20 Python-3.10.12 torch-2.1.0+cu121 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 168 layers, 11125971 parameters, 0 gradients, 28.4 GFLOPs
Class Images Instances Box(P R mAP50 mAP50-95): 100% 3/3 [00:02<00:00, 1.16it/s]
all 68 161 0.771 0.627 0.686 0.379
Speed: 0.3ms pre-process, 7.8ms inference, 0.0ms loss, 3.1ms post-process per image
Results saved to runs/detect/train2

```

Figure 4.5 Training and testing process in Google Collab

4.1.2 Offline

The researcher uses command prompt, a built-in command line interpreter application available in most Windows operating systems. With the same 4 different epochs, 25, 50, 75 and 100 epochs, at 3064 datasets, the researcher trains the YOLO v8 model using CPU. However, by using CPU to train the YOLO v8 model, it took a very significant amount of time. For 25 epochs, it took the researcher 15 hours to complete the whole training and testing process and 30 hours for 50 epochs. The researcher did not continue the training and testing for 75 epochs and 100 epochs as it will damage the CPU. Therefore, the researcher opted to train the YOLO v8 model using GPU.

By using GPU to train the YOLO v8 model, the researcher was able to reduce amount of time for training and testing significantly. For 25 epochs with 3064 datasets, the training and testing process was reduced from 15 hours to 1 hour. At 50 epochs with 3064 datasets, the training and testing process was reduced from 30 hours to 1 hour and 30 minutes. Therefore, the researcher was able to continue the training and testing process for 75 and 100 epochs with 3064 datasets. Table 4.1 shows the method and time taken for training and testing for each epochs. Figure 4.6 to Figure 4.9 shows the training and testing at 25 epochs with 339 datasets process via offline.

	from	n	params	module	arguments
0	-1	1	464	ultralytics.nn.modules.conv.Conv	[3, 16, 3, 2]
1	-1	1	4672	ultralytics.nn.modules.conv.Conv	[16, 32, 3, 2]
2	-1	1	7360	ultralytics.nn.modules.block.C2f	[32, 32, 1, True]

Figure 4.6 Training and testing process via offline

	from	n	params	module	arguments
0	-1	1	464	ultralytics.nn.modules.conv.Conv	[3, 16, 3, 2]
1	-1	1	4672	ultralytics.nn.modules.conv.Conv	[16, 32, 3, 2]
2	-1	1	7360	ultralytics.nn.modules.block.C2f	[32, 32, 1, True]
3	-1	1	18560	ultralytics.nn.modules.conv.Conv	[32, 64, 3, 2]
4	-1	2	49664	ultralytics.nn.modules.block.C2f	[64, 64, 2, True]
5	-1	1	73984	ultralytics.nn.modules.conv.Conv	[64, 128, 3, 2]
6	-1	2	197632	ultralytics.nn.modules.block.C2f	[128, 128, 2, True]
7	-1	1	295424	ultralytics.nn.modules.conv.Conv	[128, 256, 3, 2]
8	-1	1	460288	ultralytics.nn.modules.block.C2f	[256, 256, 1, True]
9	-1	1	164608	ultralytics.nn.modules.block.SPPF	[256, 256, 5]
10	-1	1	0	torch.nn.modules.upsampling.Upsample	[None, 2, 'nearest']
11	[-1, 6]	1	0	ultralytics.nn.modules.conv.Concat	[1]
12	-1	1	148224	ultralytics.nn.modules.block.C2f	[384, 128, 1]
13	-1	1	0	torch.nn.modules.upsampling.Upsample	[None, 2, 'nearest']
14	[-1, 4]	1	0	ultralytics.nn.modules.conv.Concat	[1]
15	-1	1	37248	ultralytics.nn.modules.block.C2f	[192, 64, 1]
16	-1	1	36992	ultralytics.nn.modules.conv.Conv	[64, 64, 3, 2]
17	[-1, 12]	1	0	ultralytics.nn.modules.conv.Concat	[1]
18	-1	1	123648	ultralytics.nn.modules.block.C2f	[192, 128, 1]
19	-1	1	147712	ultralytics.nn.modules.conv.Conv	[128, 128, 3, 2]
20	[-1, 9]	1	0	ultralytics.nn.modules.conv.Concat	[1]
21	-1	1	493056	ultralytics.nn.modules.block.C2f	[384, 256, 1]
22	[15, 18, 21]	1	751507	ultralytics.nn.modules.head.Detect	[1, [64, 128, 256]]

Model summary: 225 layers, 3011043 parameters, 3011027 gradients

Figure 4.7 Training and testing process via offline

```

Select Administrator: Command Prompt - python solar25.py
Freezing layer 'model.22.dfl.conv.weight'
AMP: running Automatic Mixed Precision (AMP) checks with YOLOv8n...
C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv\Lib\site-packages\torch\nn\modules\conv.py
:456: UserWarning: Plan failed with a cudnnException: CUDNN_BACKEND_EXECUTION_PLAN_DESCRIPTOR: cudnnFinalize Descriptor
Failed cudnn_status: CUDNN_STATUS_NOT_SUPPORTED (Triggered internally at ..\aten\src\ATen\native\cudnn\Conv_v8.cpp:919.)
    return F.conv2d(input, weight, bias, self.stride,
AMP: checks passed
train: Scanning C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv\F1\train\labels.cache...
val: Scanning C:\Users\Outrider\OneDrive\Documents\UTEM\FYP2\yolov8-silva-main\solar25\venv\F1\valid\labels.cache... 8
Plotting labels to runs\detect\train8\labels.jpg...
optimizer: 'optimizer=auto' found, ignoring 'lr=0.01' and 'momentum=0.937' and determining best 'optimizer', 'lr' and
'momentum' automatically...
optimizer: AdamW(lr=0.002, momentum=0.9) with parameter groups 57 weight(decay=0.0), 64 weight(decay=0.0005), 63 bias(de
cay=0.0)
Image sizes 640 train, 640 val
Using 8 dataloader workers
Logging results to runs\detect\train8
Starting training for 100 epochs...

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
1/100   2.2G     nan      nan      nan      12         640: 100%|██████████| 6/6 [00:07<00:00, 1.32
Class   Images  Instances  Box(P  R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:00<0
all     8        7        0      0      0      0

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
2/100   2.23G    nan      nan      nan      9          640: 100%|██████████| 6/6 [00:06<00:00, 1.12
Class   Images  Instances  Box(P  R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:00<0
all     8        7        0      0      0      0

```

Figure 4.8 Training and testing process via offline

```

Administrator: Command Prompt - python solar25.py
'momentum' automatically...
optimizer: AdamW(lr=0.002, momentum=0.9) with parameter groups 57 weight(decay=0.0), 64 weight(decay=0.0005), 63 bias(de
cay=0.0)
Image sizes 640 train, 640 val
Using 8 dataloader workers
Logging results to runs\detect\train8
Starting training for 100 epochs...

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
1/100   2.2G     nan      nan      nan      12         640: 100%|██████████| 6/6 [00:07<00:00, 1.32
Class   Images  Instances  Box(P  R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:00<0
all     8        7        0      0      0      0

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
2/100   2.23G    nan      nan      nan      9          640: 100%|██████████| 6/6 [00:06<00:00, 1.12
Class   Images  Instances  Box(P  R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:00<0
all     8        7        0      0      0      0

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
3/100   2.23G    nan      nan      nan      15         640: 100%|██████████| 6/6 [00:06<00:00, 1.09
Class   Images  Instances  Box(P  R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:00<0
all     8        7        0      0      0      0

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
4/100   2.24G    nan      nan      nan      8          640: 100%|██████████| 6/6 [00:06<00:00, 1.08
Class   Images  Instances  Box(P  R      mAP50  mAP50-95): 100%|██████████| 1/1 [00:00<0
all     8        7        0      0      0      0

Epoch  GPU_mem  box_loss  cls_loss  dfl_loss  Instances  Size
5/100   2.23G    nan      nan      nan      17         640: 33%|███████| 2/6 [00:02<00:04, 1.12

```

Figure 4.9 Training and testing process via offline

Table 4.1 Method and time taken for training and testing

Method	No. of Dataset	Time Taken (25 Epochs)	Time Taken (50 Epochs)	Time Taken (75 Epochs)	Time Taken (100 Epochs)
Google Collab	339	20 Minutes	40 Minutes	60 Minutes	80 Minutes
CPU (offline)	3064	15 Hours	30 Hours	N/A	N/A
GPU (offline)	3064	60 Minutes	90 Minutes	120 Minutes	150 Minutes

اونيورسيتي تيكنيكل مليسيا ملاك

4.2 Confusion Matrix and Graph

A confusion matrix is a simple table used to evaluate the performance of a classification model. It shows the actual versus predicted values, giving the researcher a clear idea of how well the model is identifying each class. In the matrix, true positives (TP) and true negatives (TN) represent correct predictions, whereas false positives (FP) and false negatives (FN) represent errors. It is a fundamental tool for understanding a model's predictive accuracy, allowing us to measure metrics like precision and recall. Table 4.2 shows the general confusion matrix. Below explains the definition of true positive (TP), false positive (FP), true negative (TN) and false negative (FN).

- True Positives (TP): The cases in which the model correctly predicted the positive class. For example, there is defect on solar panel and the developed system also detects there is defect on solar panel

- True Negatives (TN): The cases in which the model correctly predicted the negative class. For example, there is no defect on solar panel and the developed system also detects there is no defect on solar panel.
- False Positives (FP): The cases in which the model incorrectly predicted the positive class (also known as Type I error). For example, there is defect on solar panel, but the developed system detects there is no defects on solar panel.
- False Negatives (FN): The cases in which the model incorrectly predicted the negative class (also known as Type II error). For example, there is no defect on solar panel, but the developed system detects there is defect on solar panel.

The evaluation of the model's performance involves calculating key metrics: accuracy, precision, recall, and the F1 score. These metrics are derived from the confusion matrix, a tool that summarizes the model's predictions, allowing for a detailed analysis of its ability to correctly identify defects. Accuracy measures the overall correctness of the model, precision assesses its ability to identify true positives out of all positive predictions, recall evaluates the model's capability to detect true positives from the actual positives, and the F1 score provides a balance between precision and recall, offering a single measure of the model's efficiency.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

$$F1 \text{ Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (6)$$

Table 4.2 Confusion matrix

Confusion Matrix		Actual Class	
		Positive	Negative
Predicted Class	Positive	True Positive (TP)	True Negative (TN)
	Negative	False Positive (FP)	False Negative (FN)

4.2.1 Google Collab

As mentioned previously, in FYP 1 the YOLO v8 model is trained and tested using Google Collab. The researcher trains and tests at 4 different epochs, 25, 50, 75 and 100 epochs with 339 datasets. All results were automatically generated and saved in Google Collab. The researcher also saved all the results in PC.

Figure 4.10 shows that at TP, 65% of the time the model correctly detects the hotspot as a hotspot while at FP, 35% of the time the model falsely detects the hotspot as a background. At TN, 100% of the time the model correctly detects the background as a background while at FN, 0% of the time the model falsely detects a hotspot even though there is no hotspot. Figure 4.11 shows graph results for the model. However, the researcher focuses on the train/box loss and train/cls loss graph. Both graphs show the desired gradient however there are still room for improvement. Both matrix and graphs are the result of the model trained at 25 epochs with 339 datasets via Google Collab.

Figure 4.12 shows that 69% of the time the model correctly detects the hotspot as a hotspot while 31% of the time the model falsely detects the hotspot as a background. Figure 4.13 shows graph results for the model. At TN, 100% of the time the model correctly detects the background as a background while at FN, 0% of the time the model falsely detects a hotspot even though there is no hotspot. Both train/box loss and train/cls loss graphs show the desired gradient however there are still room for improvement. Both matrix and graphs are the result of the model trained at 50 epochs with 339 datasets via Google Collab.

Figure 4.14 shows that 67% of the time the model correctly detects the hotspot as a hotspot while 33% of the time the model falsely detects the hotspot as a background. Figure 4.15 shows graph results for the model. At TN, 100% of the time the model correctly detects the background as a background while at FN, 0% of the time the model falsely detects a hotspot even though there is no hotspot. Both train/box loss and train/cls loss graphs show the desired gradient however there are still room for improvement. Both matrix and graphs are the result of the model trained at 75 epochs with 339 datasets via Google Collab.

Figure 4.16 shows that 68% of the time the model correctly detects the hotspot as a hotspot while 32% of the time the model falsely detects the hotspot as a background. Figure 4.17 shows graph results for the model. At TN, 100% of the time the model correctly detects the background as a background while at FN, 0% of the time the model falsely detects a hotspot even though there is no hotspot. Both train/box loss and train/cls loss graphs show the desired gradient however there are still room for improvement. Both matrix and graphs are the result of the model trained at 100 epochs with 339 datasets via Google Collab. Table 4.3 shows all the results for all epochs with 339 datasets via Google Collab. Table 4.4 shows the number of images detected for all epochs with 339 datasets via Google Collab.

By obtaining the confusion matrix for all epochs, the researcher can begin evaluating the AI model's accuracy, precision, recall and F1 scorer for each epoch. Table 4.5 shows the performance metrics for all epochs with 339 datasets via Goggle Collab. The AI model has the highest accuracy of 84.5% at 50 epochs with 339 datasets. This contradicts the reseacher's hypothesis that the more extensive training the AI model undergoes, the higher the accuracy of the AI model.

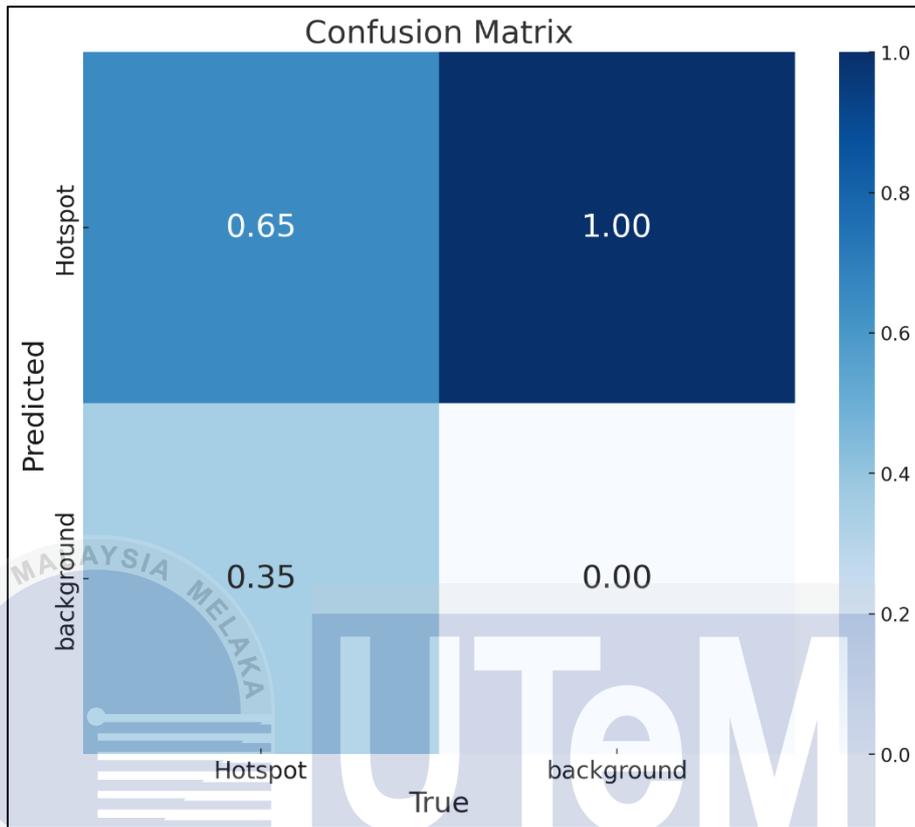


Figure 4.10 Confusion matrix at 25 epochs (Google Collab)

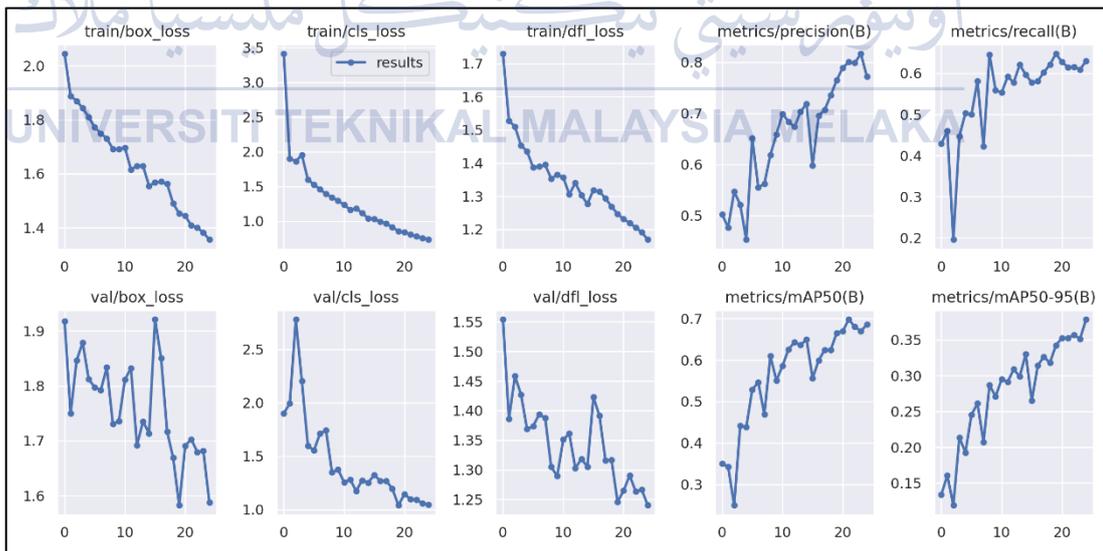


Figure 4.11 Results graph at 25 epochs (Google Collab)

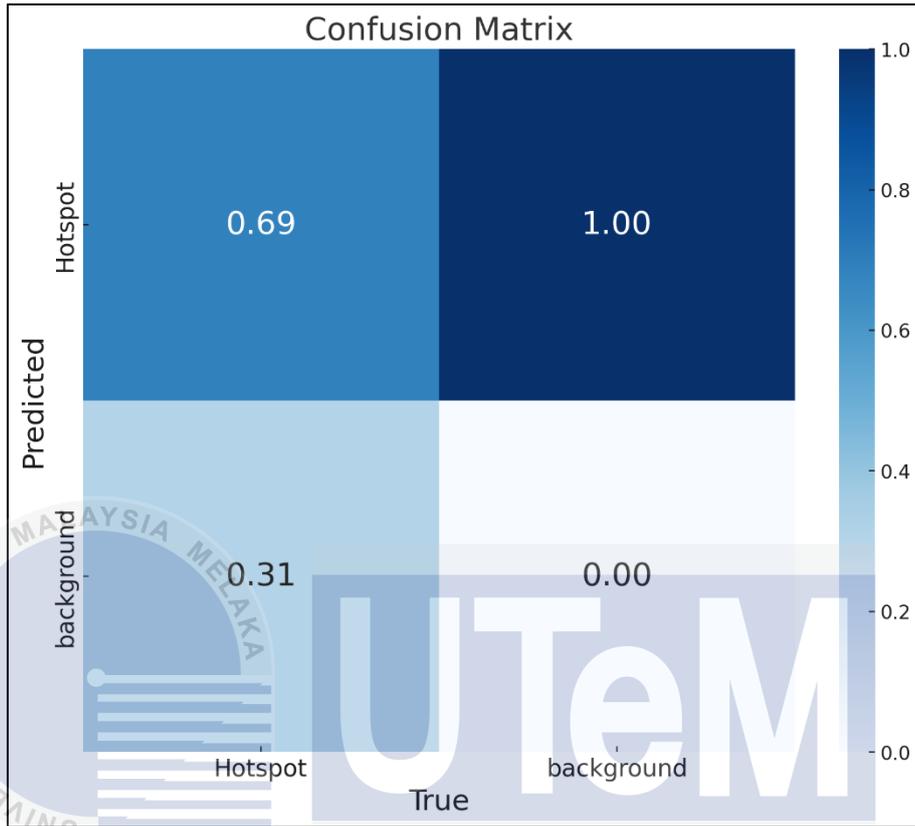


Figure 4.12 Confusion matrix at 50 epochs (Google Collab)

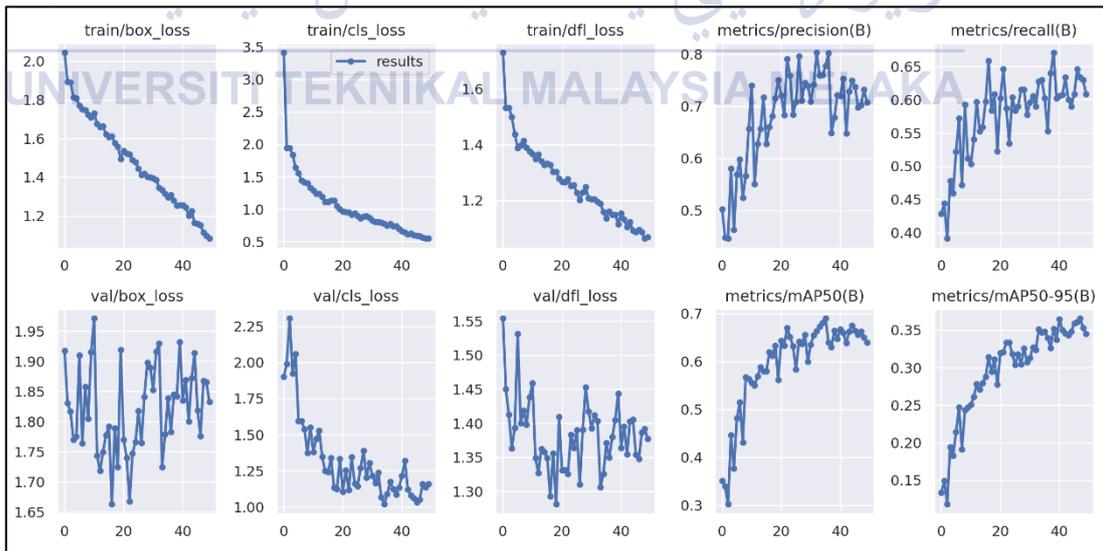


Figure 4.13 Results graph at 50 epochs (Google Collab)

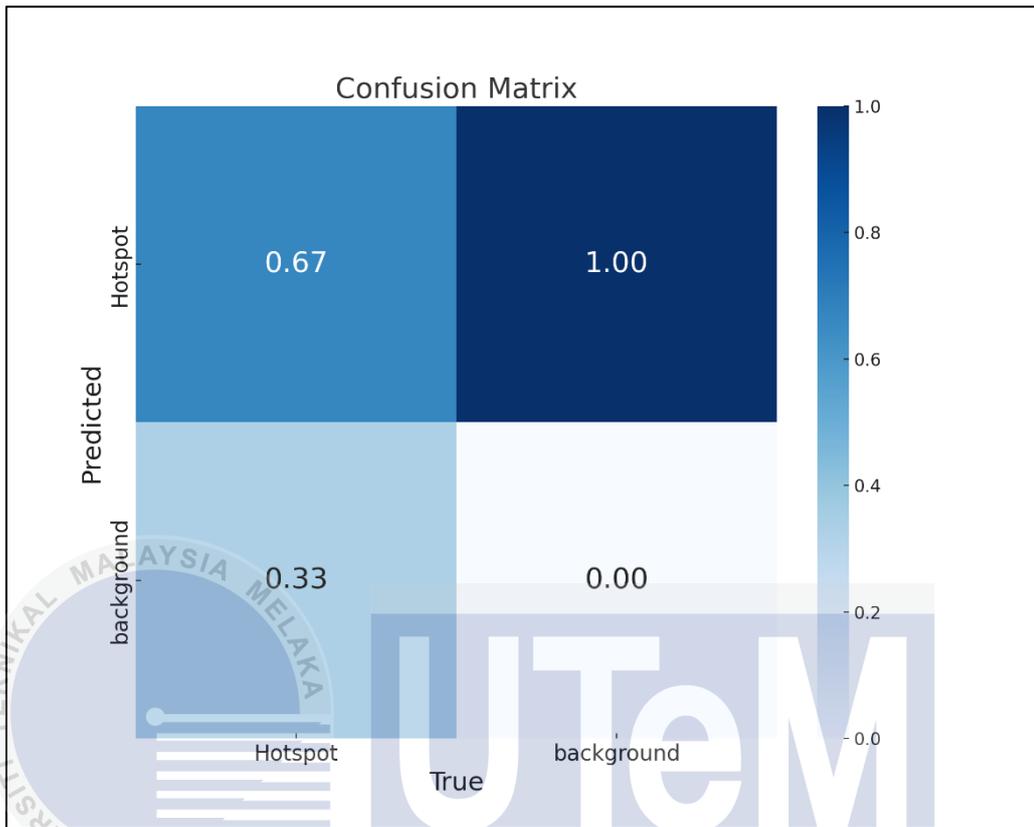


Figure 4.14 Confusion matrix at 75 epochs (Google Collab)

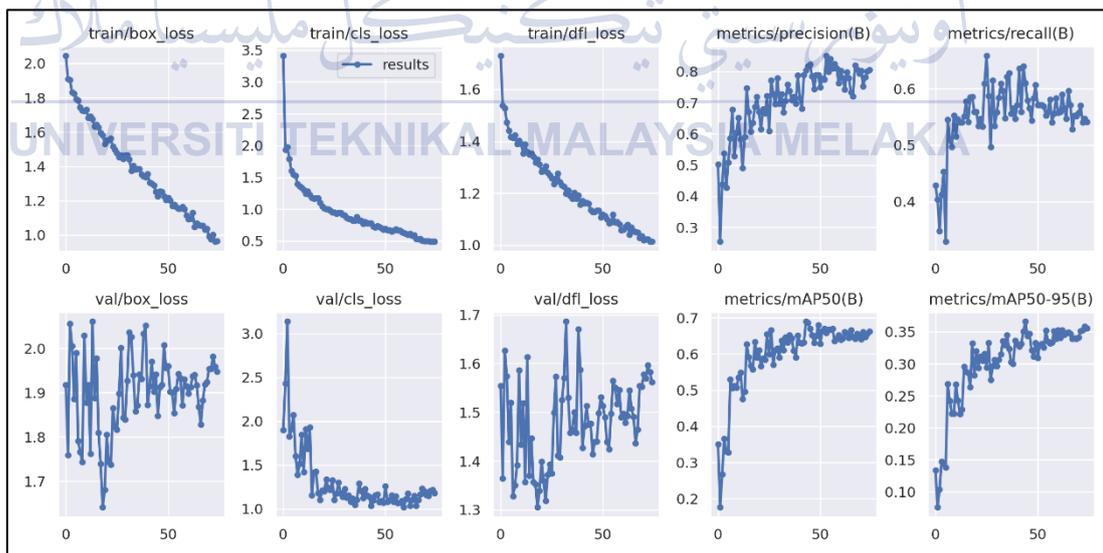


Figure 4.15 Results graph at 75 epochs (Google Collab)

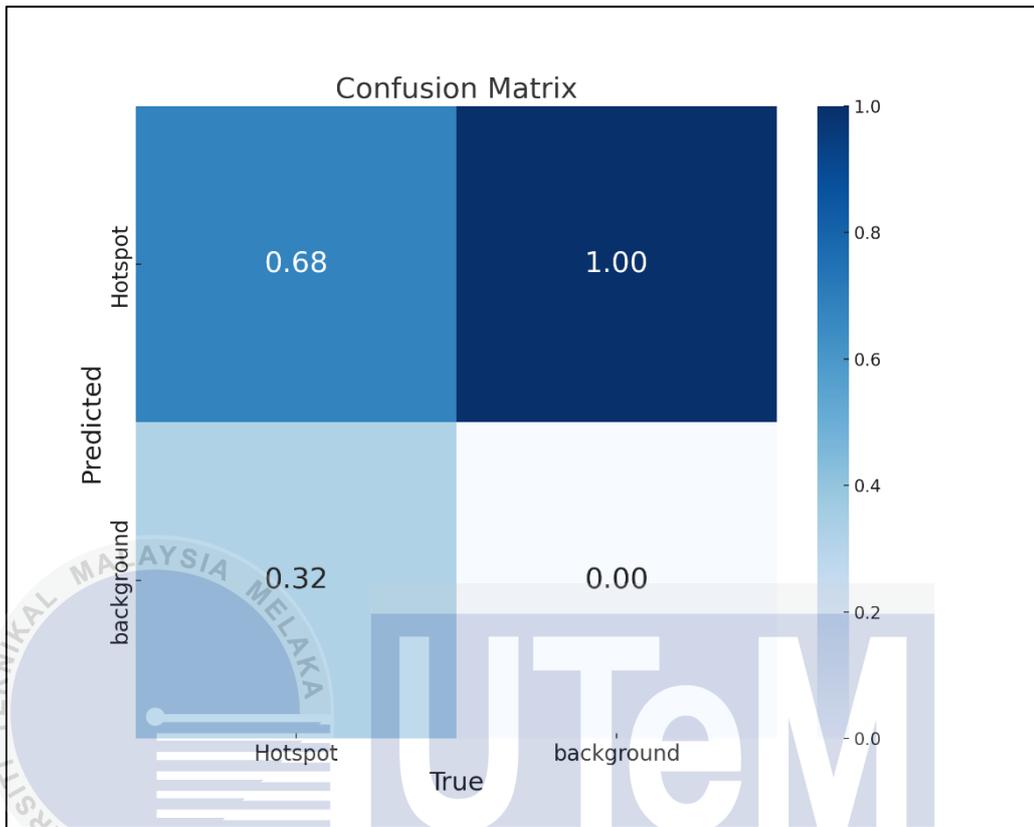


Figure 4.16 Confusion matrix at 100 epochs (Google Collab)

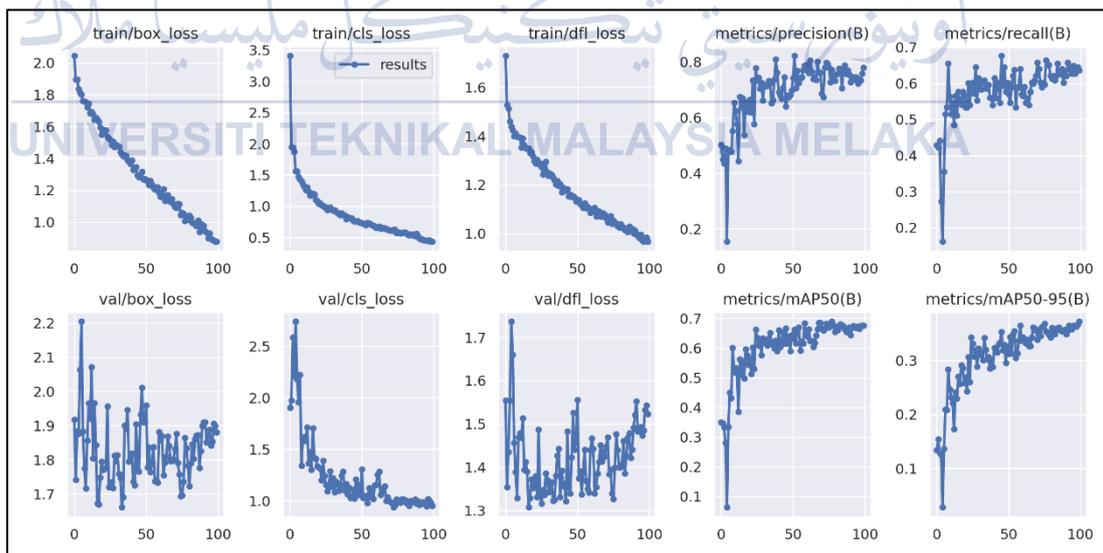


Figure 4.17 Results graph at 100 epochs (Google Collab)

Table 4.3 Results for all epochs (Google Collab)

Number of Epochs	TP	FP	TN	FN
25	0.65	0.35	1.00	0.00
50	0.69	0.31	1.00	0.00
75	0.67	0.33	1.00	0.00
100	0.68	0.32	1.00	0.00

Table 4.4 Number of images detected for all epochs (Google Collab)

Number of Epochs	TP	FP	TN	FN
25	220	119	339	0
50	233	106	339	0
75	227	112	339	0
100	231	108	339	0

Table 4.5 Performance metrics for all epochs (Google Collab)

Number of Epochs	Accuracy	Precision	Recall	F1 Score
25	0.825	0.65	1.0	0.7879
50	0.845	0.69	1.0	0.8166
75	0.835	0.67	1.0	0.8024
100	0.84	0.68	1.0	0.8095

4.2.2 Offline

In FYP 2, the AI model is trained and tested via offline using CPU and GPU at 4 different epochs, 25, 50, 75 and 100 epochs with 3064 datasets. Both CPU and GPU shows the same result as it was trained and tested on the same number of epochs and datasets. The only difference is the time taken to complete the whole process. All results were automatically generated and saved in the PC.

Figure 4.18 shows that at TP, 93% of the time the model correctly detects the hotspot as a hotspot while at FP, 7% of the time the model falsely detects the hotspot as a background. At TN, 100% of the time the model correctly detects the background as a background while at FN, 0% of the time the model falsely detects a hotspot even though there is no hotspot. Figure 4.19 shows graph results for the model. However, as mention previously, the researcher focuses on the train/box loss and train/cls loss graph. Both graphs show the desired gradient however there are still room for improvement. Both matrix and graphs are the result of the model trained at 25 epochs with 3064 datasets via offline.

Figure 4.20 shows that 92% of the time the model correctly detects the hotspot as a hotspot while 8% of the time the model falsely detects the hotspot as a background. Figure 4.21 shows graph results for the model. At TN, 100% of the time the model correctly detects the background as a background while at FN, 0% of the time the model falsely detects a hotspot even though there is no hotspot. Both train/box loss and train/cls loss graphs show the desired gradient however there are still room for improvement. Both matrix and graphs are the result of the model trained at 50 epochs with 3064 datasets via offline.

Figure 4.22 shows that 92% of the time the model correctly detects the hotspot as a hotspot while 8% of the time the model falsely detects the hotspot as a background. Figure 4.23 shows graph results for the model. At TN, 100% of the time the model correctly detects the background as a background while at FN, 0% of the time the model falsely detects a hotspot even though there is no hotspot. Both train/box loss and train/cls loss graphs show the desired gradient however there are still room for improvement. Both matrix and graphs are the result of the model trained at 75 epochs with 3064 datasets via offline.

Figure 4.24 shows that 92% of the time the model correctly detects the hotspot as a hotspot while 8% of the time the model falsely detects the hotspot as a background. Figure 4.25 shows graph results for the model. At TN, 100% of the time the model correctly detects the background as a background while at FN, 0% of the time the model falsely detects a hotspot even though there is no hotspot. Both train/box loss and train/cls loss graphs show the desired gradient however there are still room for improvement. Both matrix and graphs are the result of the model trained at 100 epochs with 3064 datasets via offline. Table 4.6 shows all the results for all epochs with 3064 datasets via offline. Table 4.7 shows the number of images detected for all epochs with 3064 datasets via offline.

By obtaining the confusion matrix for all epochs, the researcher can begin evaluating the AI model's accuracy, precision, recall and F1 scorer for each epoch. Table 4.8 shows the performance metrics for all epochs with 3064 datasets via offline. The AI model at 50, 75 and 100 epochs with 3064 datasets have the same accuracy of 96%. The AI model at 25 epochs with 3064 datasets has a slightly higher accuracy of 96.5%. This slightly contradicts the researcher's hypothesis that the more extensive training the AI model undergoes, the higher the accuracy of the AI model.

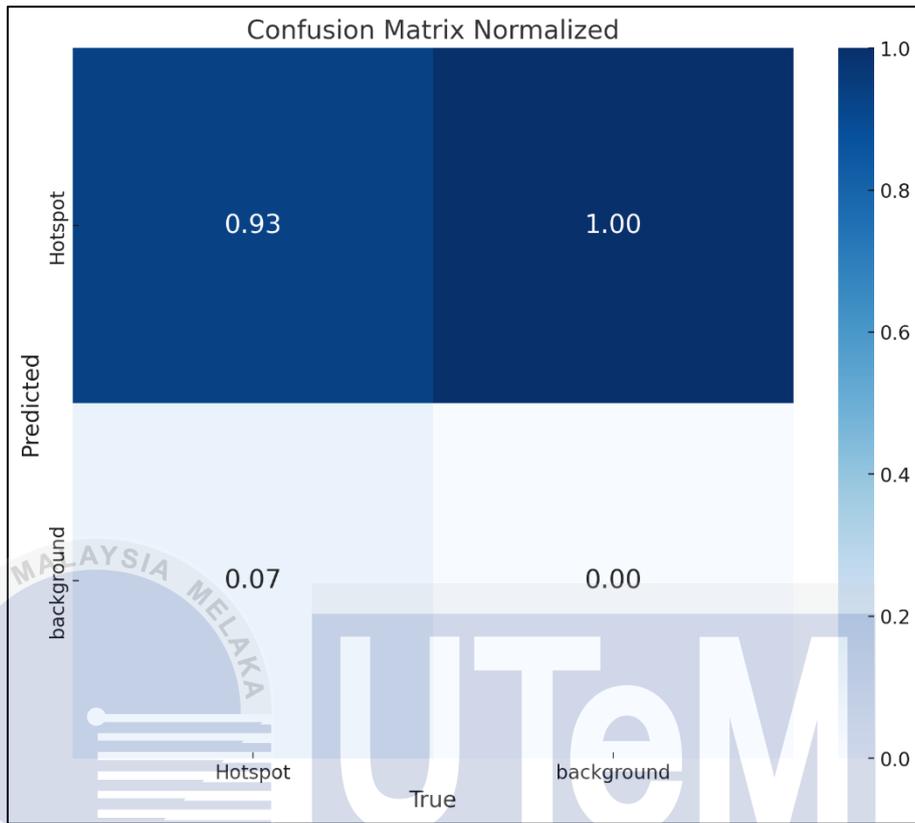


Figure 4.18 Confusion matrix at 25 epochs (Offline)

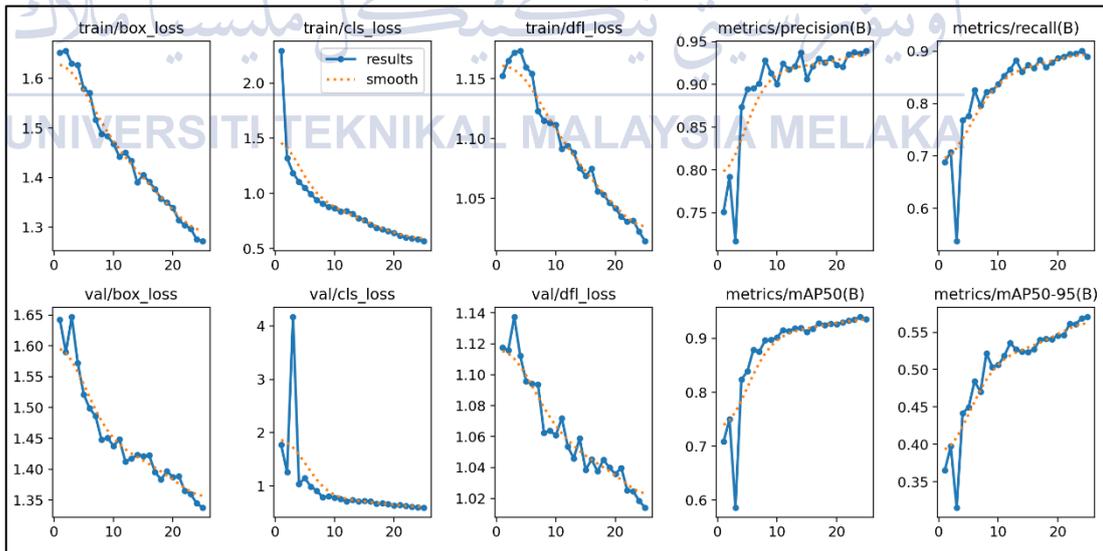


Figure 4.19 Results graph at 25 epochs (Offline)

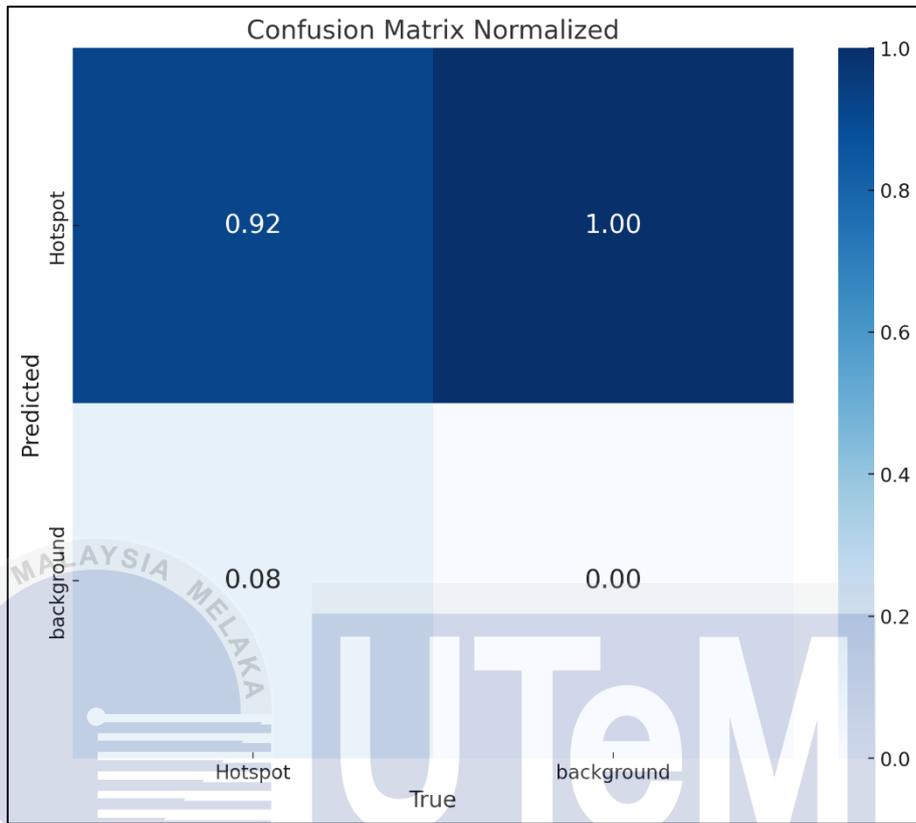


Figure 4.20 Confusion matrix at 50 epochs (Offline)

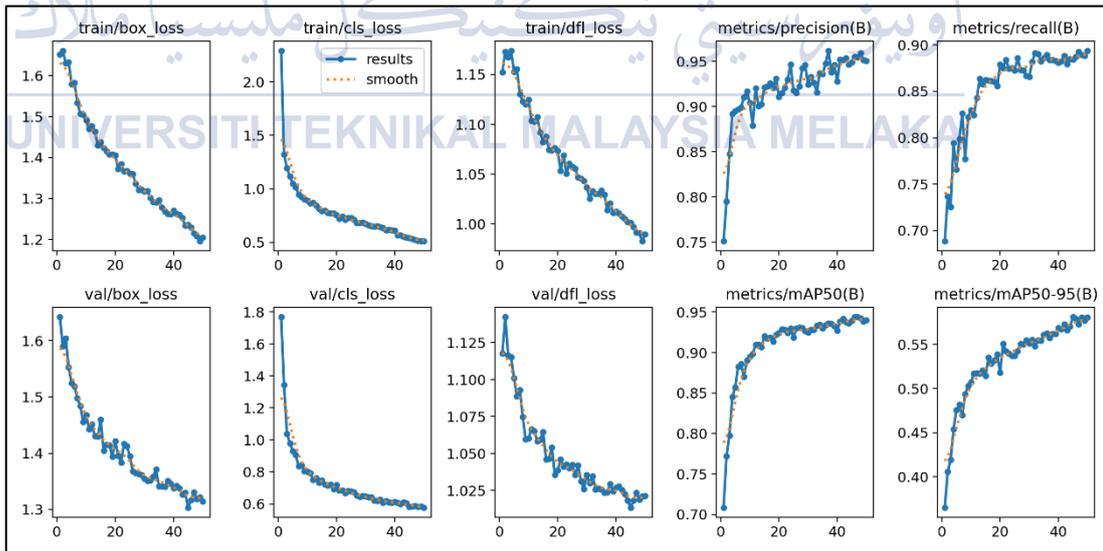


Figure 4.21 Results graph at 50 epochs (Offline)

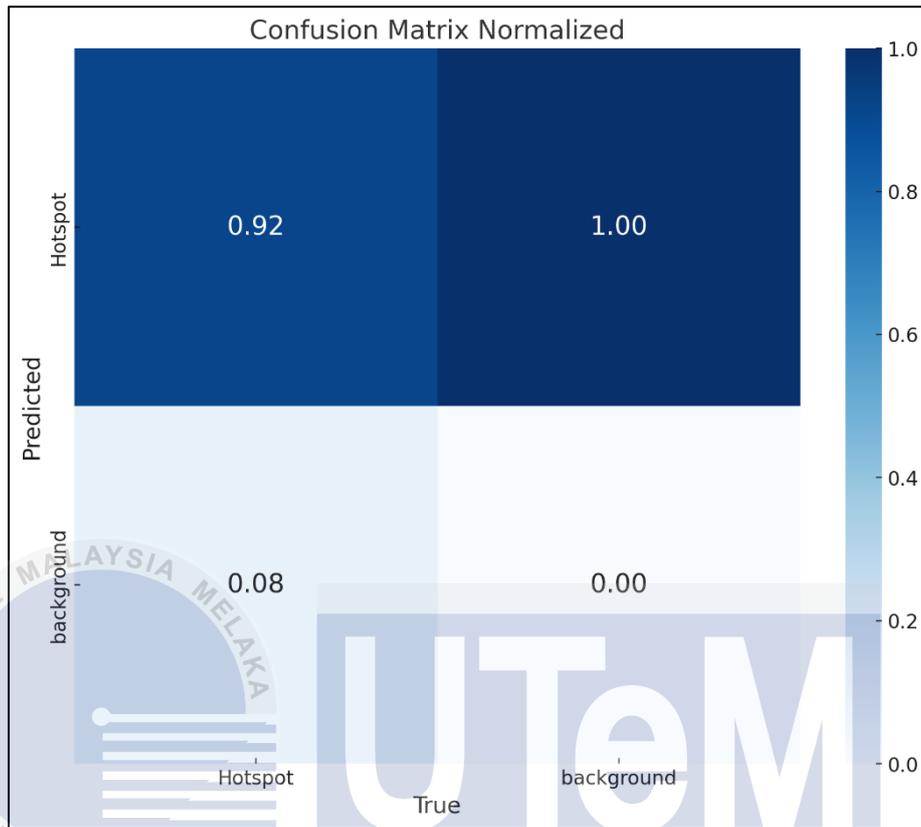


Figure 4.22 Confusion matrix at 75 epochs (Offline)

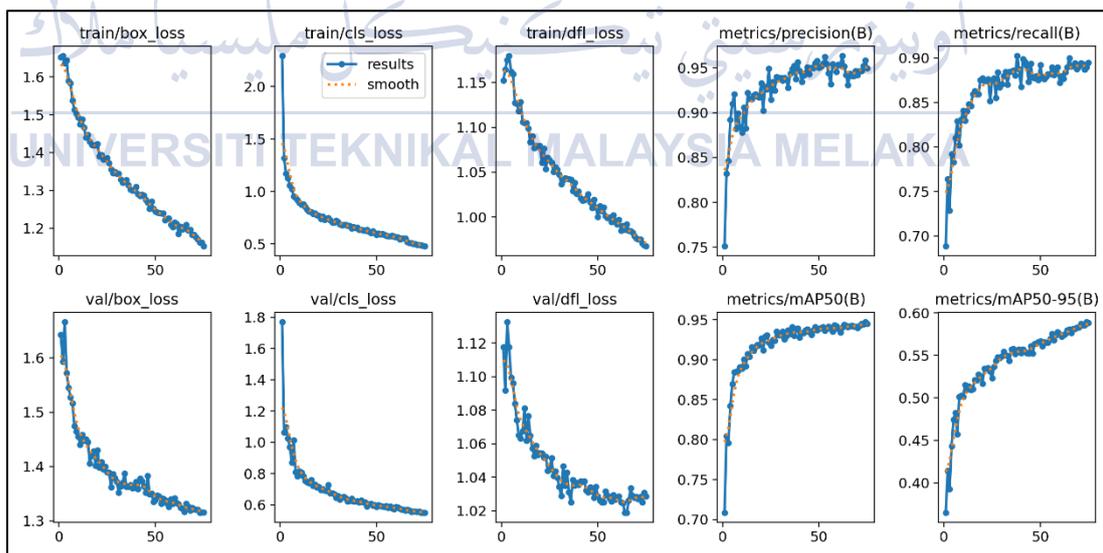


Figure 4.23 Results graph at 75 epochs (Offline)

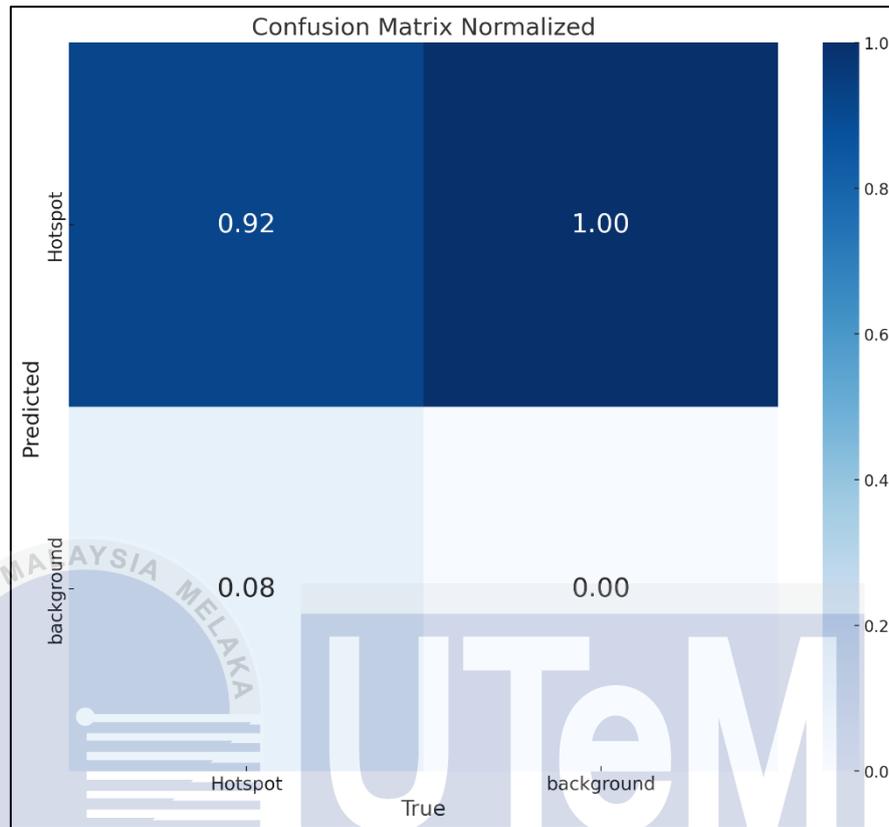


Figure 4.24 Confusion matrix at 100 epochs (Offline)

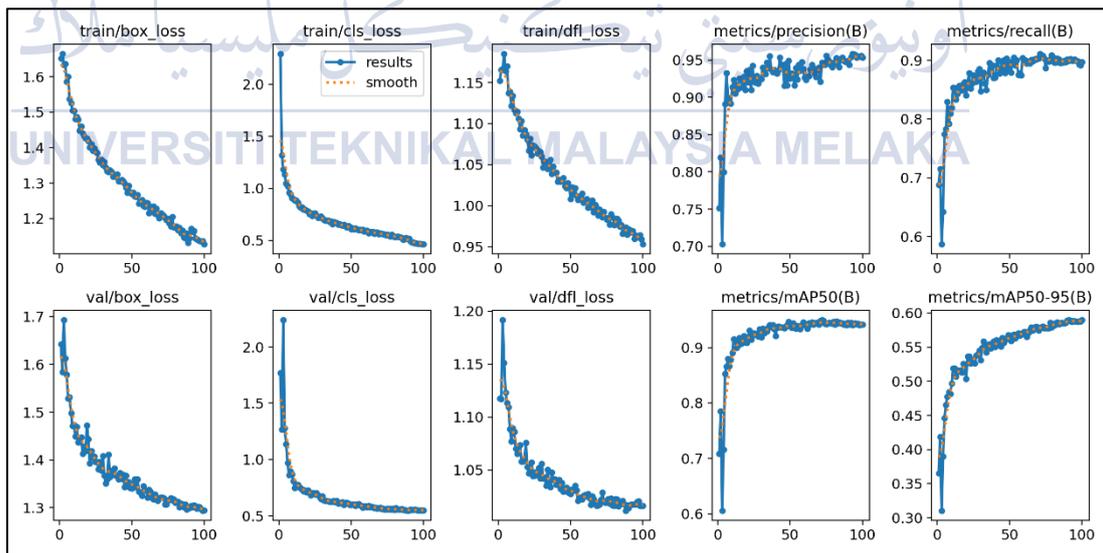


Figure 4.25 Results graph at 100 epochs (Offline)

Table 4.6 Results for all epochs (offline)

Number of Epochs	TP	FP	TN	FN
25	0.93	0.07	1.00	0.00
50	0.92	0.08	1.00	0.00
75	0.92	0.08	1.00	0.00
100	0.92	0.08	1.00	0.00

Table 4.7 Number of images detected for all epochs (offline)

Number of Epochs	TP	FP	TN	FN
25	2849	215	3064	0
50	2819	245	3064	0
75	2819	245	3064	0
100	2819	245	3064	0

Table 4.8 Performance metrics for all epochs (offline)

Number of Epochs	Accuracy	Precision	Recall	F1 Score
25	0.965	0.93	1.0	0.964
50	0.96	0.92	1.0	0.958
75	0.96	0.92	1.0	0.958
100	0.96	0.92	1.0	0.958

4.3 Validation and Inference Results

Validation is an important stage in the machine learning pipeline that allows the researcher to evaluate the quality of the AI models. Val mode in Ultralytics YOLO v8 offers a comprehensive set of tools and metrics for assessing the effectiveness of the object detection algorithms. By using YOLO v8 Val mode, the researcher can get accurate metrics like mAP50, mAP75 and mAP50-95 to comprehensively evaluate the AI model. YOLO v8 Val mode utilizes built-in features that remember training and testing settings thus simplifying the validation process. The researcher can validate the AI model with the same or different datasets and image size. By using the validation metrics, the researcher can further fine-tune the AI model for better performance. Figure 4.26 shows the code for YOLO v8 Val mode.

Inference is the process of applying the learned AI model to fresh, unobserved data. At this point, the model is put to use and implemented in practical applications. Figure 4.27 shows the code for inference custom model where it randomly chose images for hotspots detection. By applying 25% confidence level, the AI model will show any hotspots detection above 0.25.

```
[ ] from ultralytics import YOLO
# Load a model
model = YOLO("yolov8n.pt") # load an official model
model = YOLO("{HOME}/runs/detect/train/weights/best.pt") # load a custom model

# Validate the model
metrics = model.val() # no arguments needed, dataset and settings remembered
metrics.box.map # map50-95
metrics.box.map50 # map50
metrics.box.map75 # map75
metrics.box.maps # a list contains map50-95 of each category

!yolo task=detect mode=val model={HOME}/runs/detect/train/weights/best.pt data={dataset.location}/data.yaml
```

Figure 4.26 Validation code

```
[ ] from ultralytics import YOLO

# Load a model
model = YOLO("yolov8n.pt") # load an official model
model = YOLO("{HOME}/runs/detect/train/weights/best.pt") # load a custom model

# Predict the model
!yolo task=detect mode=predict model={HOME}/runs/detect/train/weights/best.pt
conf=0.25
source={dataset.location}/test/images
save=True
```

Figure 4.27 Inference code

4.3.1 Google Collab

This section is to showcase the validation and inference custom model result for 25, 50, 75 and 100 epochs with 339 datasets via Google Collab. Figure 4.28 to 4.31 shows the results for 25, 50, 75 and 100 epochs respectively.

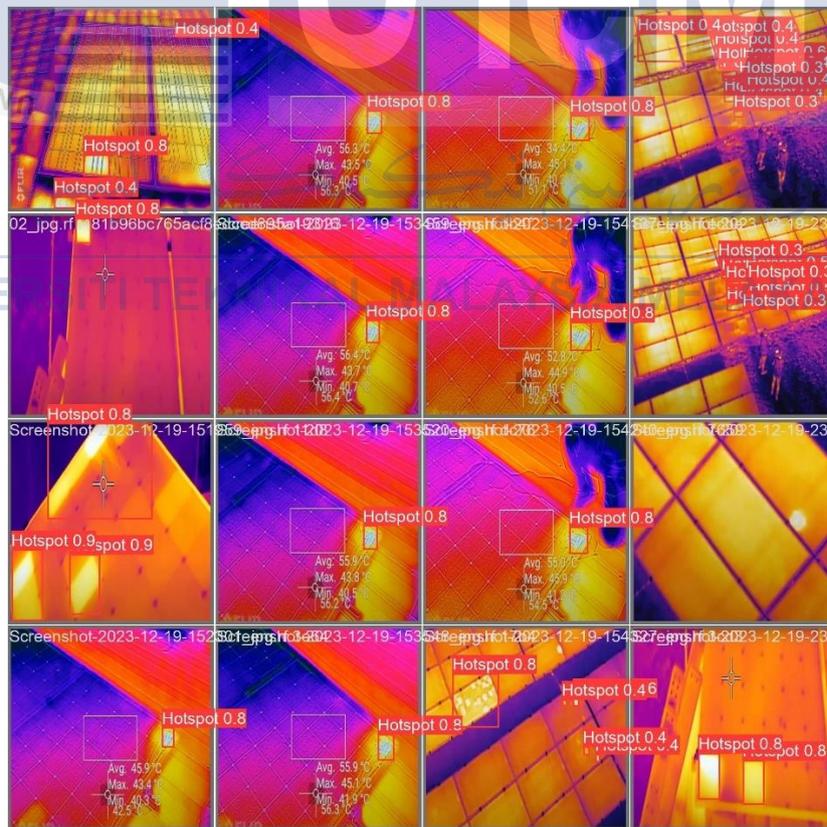


Figure 4.28 Result at 25 epochs (Google Collab)

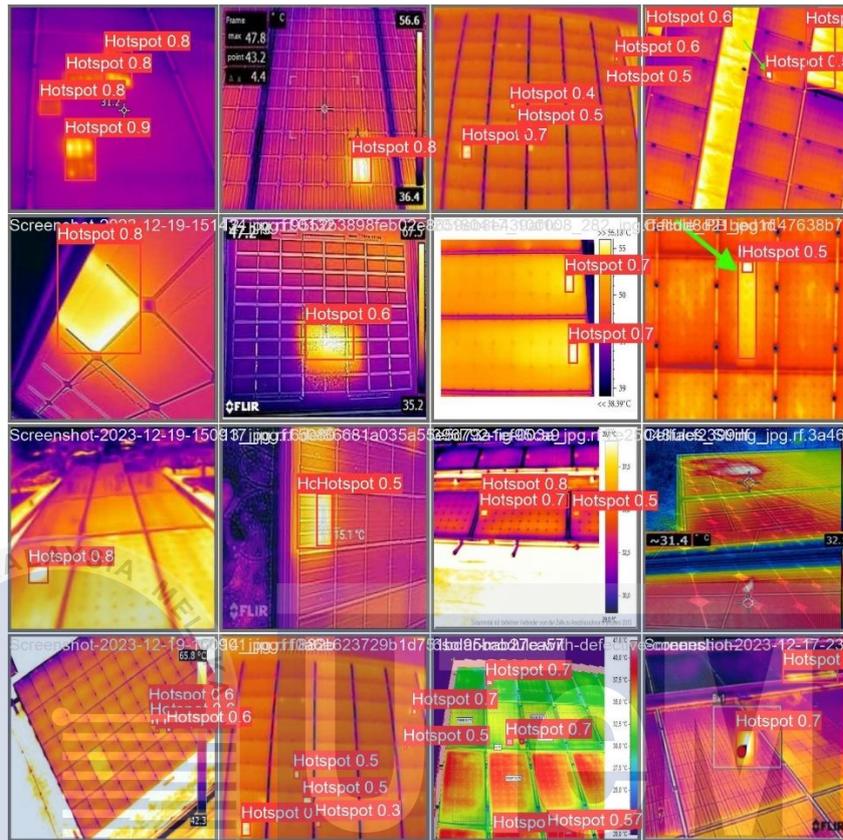


Figure 4.29 Result at 50 epochs (Google Collab)

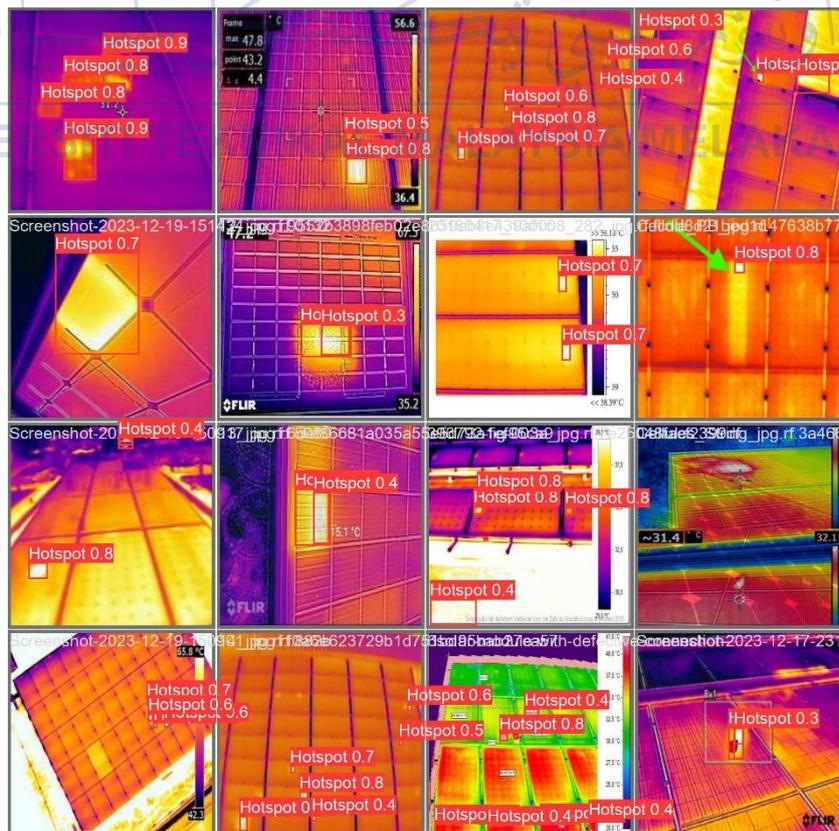


Figure 4.30 Result at 75 epochs (Google Collab)



Figure 4.31 Result at 100 epochs (Google Collab)

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.3.2 Offline

This section is to showcase the validation and inference custom model result for 25, 50, 75 and 100 epochs with 3064 datasets via offline. Figure 4.32 to 4.35 shows the results for 25, 50, 75 and 100 epochs respectively.

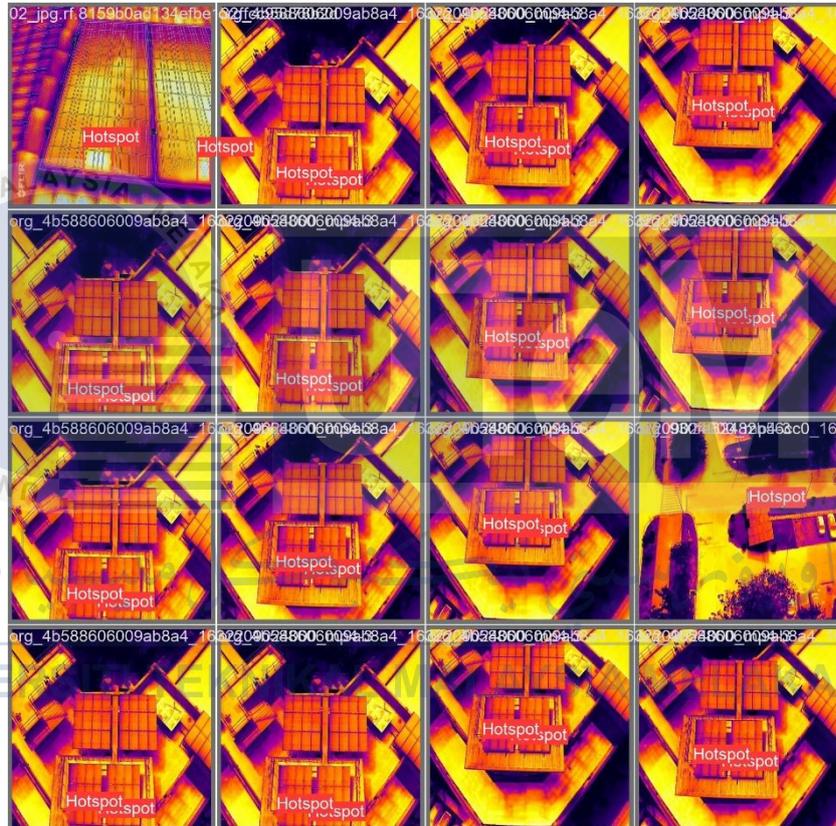


Figure 4.32 Result at 25 epochs (Offline)

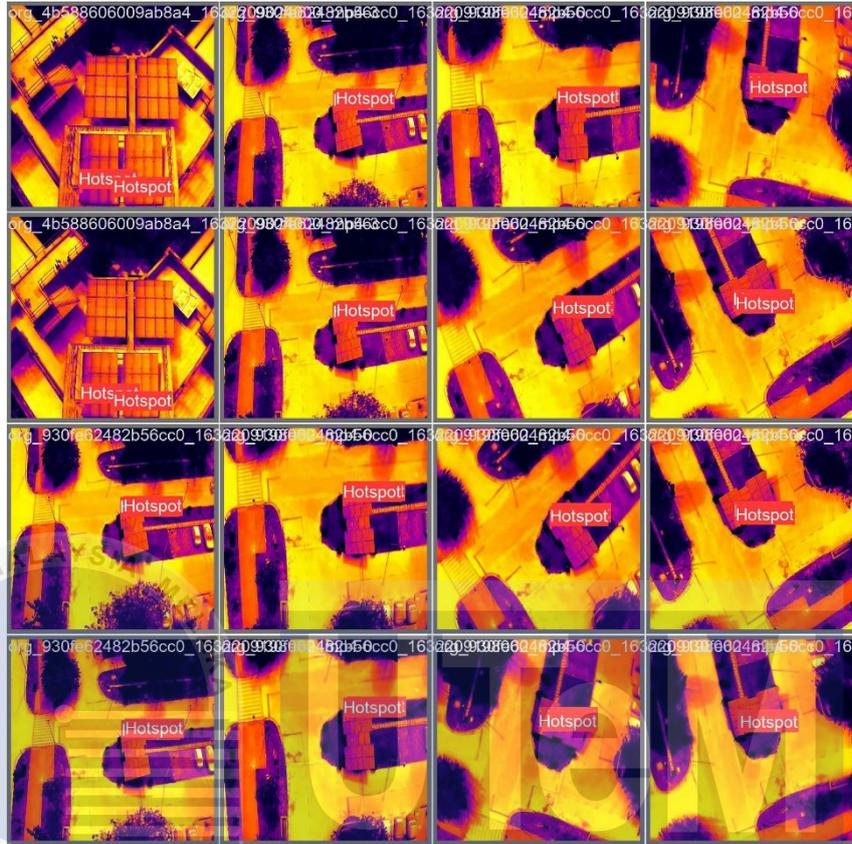


Figure 4.33 Result at 50 epochs (Offline)

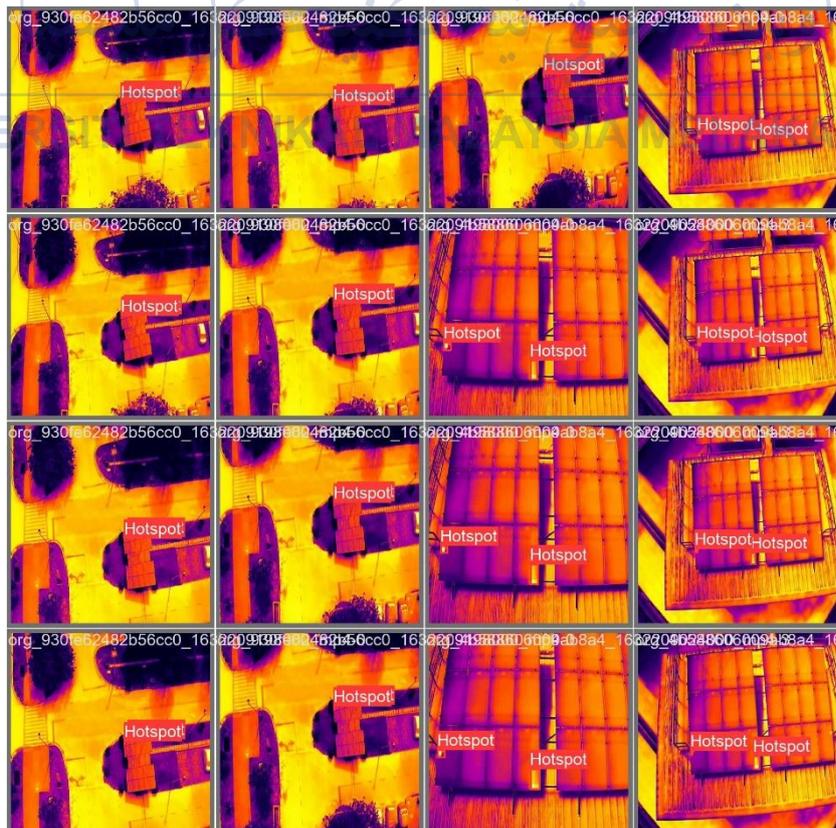


Figure 4.34 Result at 75 epochs (Offline)

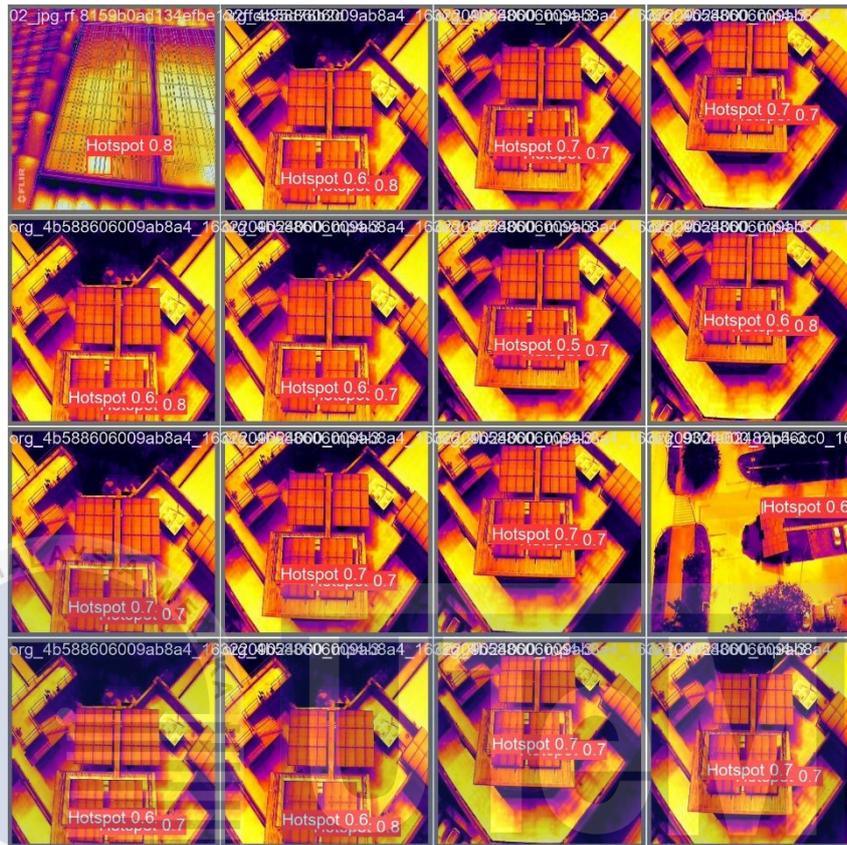


Figure 4.35 Result at 100 epochs (Offline)

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.4 Comparison Between Epochs and Dataset

This section investigates the effects of different training epochs and the number of datasets on the model's hotspot detection capability in solar panels. The analysis analyses photos processed after 25, 50, 75, and 100 epochs to determine the best training length for improved detection accuracy and fewer false positives. The analysis also analyses photos processed from datasets of sizes 339 and 3064 to determine whether a larger dataset leads to better performance. This comparison reveals the model's learning development and efficiency at various training phases, providing useful insights into optimizing deep learning procedures for better fault detection in photovoltaic systems.

4.4.1 Comparison Between Epochs

In both FYP 1 and FYP 2, the researcher trains and tests the AI model at four different epochs, 25, 50, 75 and 100 epochs. This is to determine the best training length for improved detection accuracy and fewer false positives. In FYP 1, the researcher used 339 datasets to train and test the AI model. Once the training and testing process completed, the researcher begins the validation and inference process. The researcher applies the AI model with unseen images for practical application. Figure 4.36 shows the comparison between epoch with 339 datasets. In FYP 1, the AI model at 50 epochs achieved the highest accuracy with 84.5%. However, the AI model at 100 epochs shows the highest confidence with 0.89 and 0.83. However, when compare the gap between the confidence level, 75 epochs shows the least gap between the confidence level, 0.81 and 0.83. This shows that the AI model at 75 epochs produced a more consistent result while achieving a relatively high confidence level. Thus, 75 epochs shows the most desirable result. This contradicts the researcher's hypothesis that the more extensive training and testing the AI model undergoes leads to the higher the accuracy and the higher the confidence level.

In FYP 2, the researcher used 3064 datasets to train and test the AI model. The researcher also undergoes the validation and inference process. The researcher applies the AI model with unseen images for practical application. Figure 4.37 shows the comparison between epoch with 3064 datasets. In FYP 2, the AI model at 20 epochs achieved the highest accuracy with 96.5%. However, the AI model at 100 epochs shows the highest confidence with 0.79 and 0.72. However, when compare the gap between the confidence level, the same occurrence happens. The AI model at 75 epochs shows the least gap between the confidence level, 0.74 and 0.76. This also shows that the AI model at 75 epochs produced a more consistent result while achieving a relatively high confidence level. Thus, 75 epochs shows the most desirable result. This also contradicts the researcher's hypothesis that the more extensive training and testing the AI model undergoes leads to the higher the accuracy and the higher the confidence level.

As a conclusion, in FYP 1 and FYP 2, both 75 epochs achieved the most desirable result, having the least gap between confidence while achieving a relatively high confidence level, leading to a consistent result.

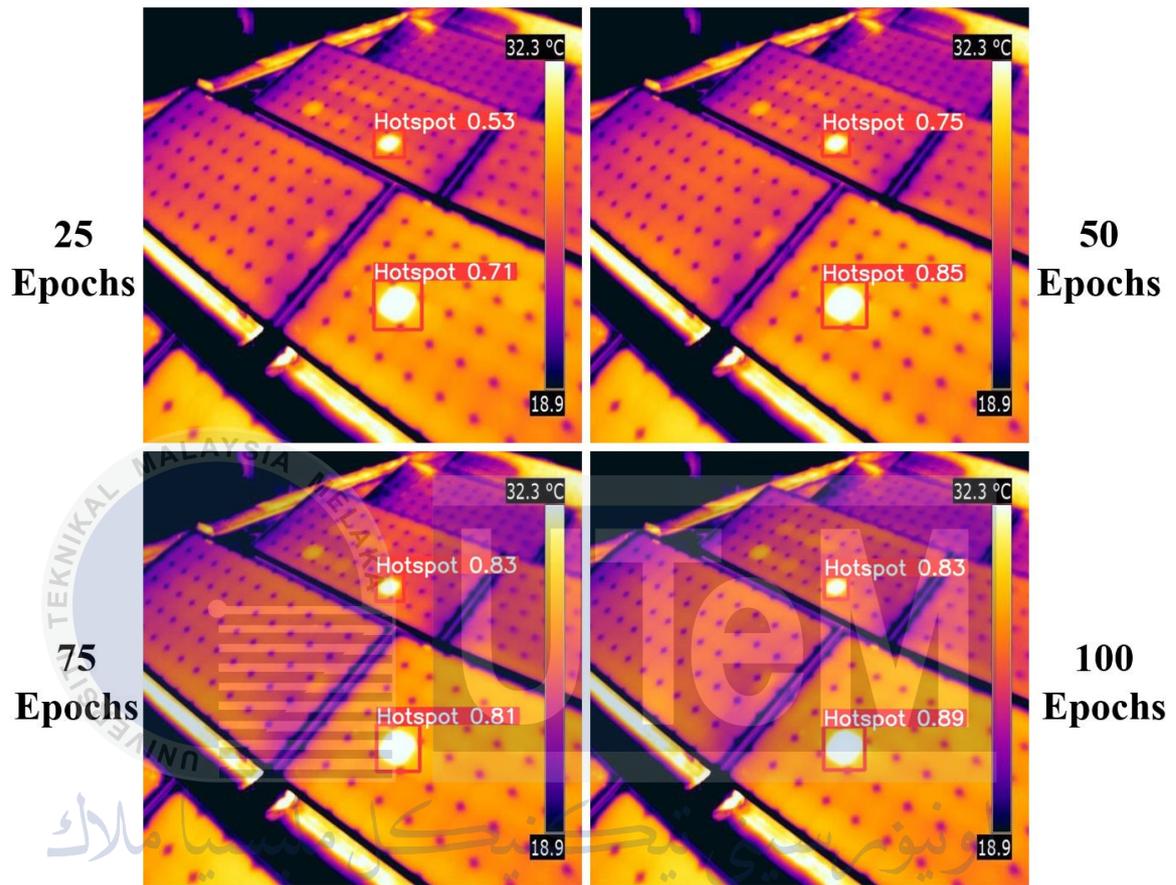


Figure 4.36 Comparison between epoch with 339 datasets

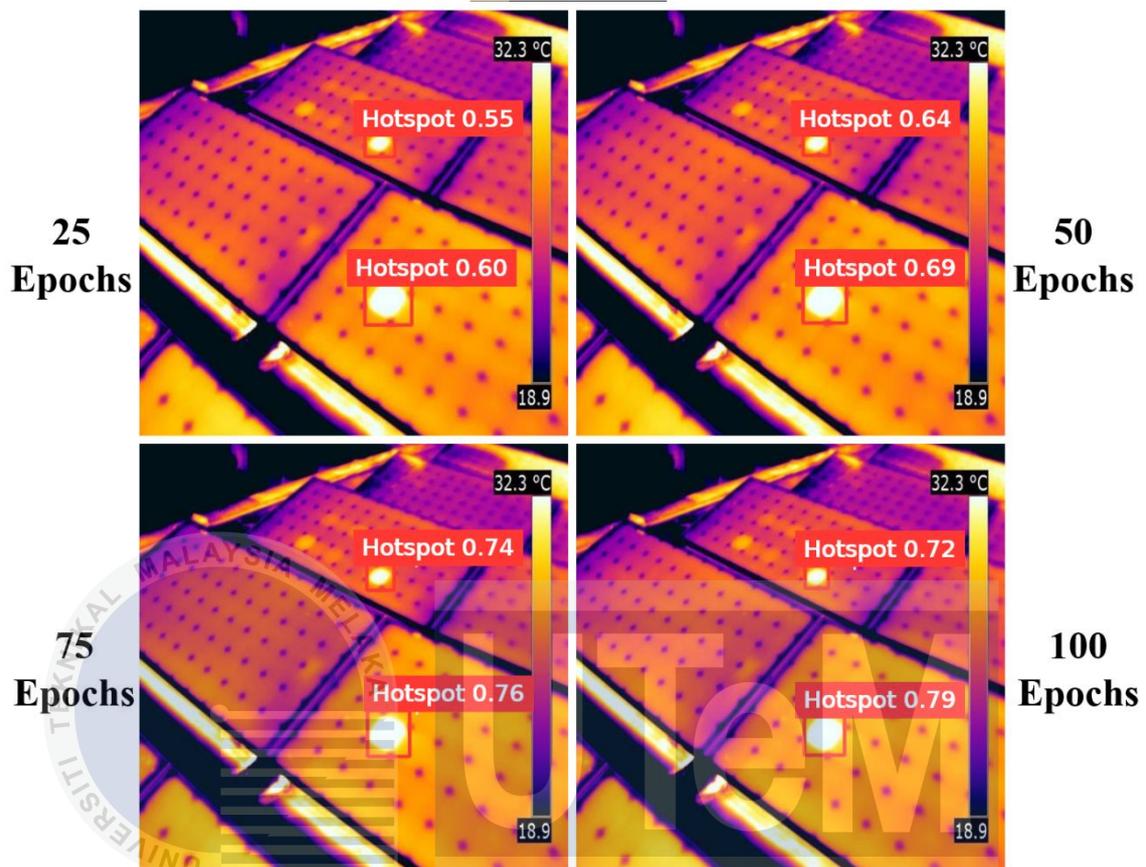


Figure 4.37 Comparison between epoch with 3064 datasets

4.4.2 Comparison Between Size of Datasets

This section the resesarcher compares the effect of the size of datasets on the performance of the AI model while training and testing at the same number of epochs. As mention previously, at 75 epochs the AI model produced the most desirable result in both FYP 1 and FYP 2, thus the researcher compares the performance of the AI model at 75 epochs with 339 datasets and 3064 datatsets. Table 4.9 shows the confusion matrix for both 75 epochs with 339 datasets and 3064 datasets. The AI model with 3064 datasets has a significantly higher true positive (TP) and lower false positive (FP) value than the AI model with 339 datasets. This shows that the AI model with 3064 datasets is able to detect more hotspot correctly than the AI model with 339 datasets.

Table 4.10 shows the overall performance of the AI model at 75 epocsh with 339 datasets and 3064 datasets. The AI model with 3064 datasets achieved a significantly higher accuracy, precision and recall value than the AI model with 339 datasets. This suggests that

AI model with 3064 datasets has a better overall performance than the AI model at 339 datasets.

Figure 4.38 and 4.39 shows the performance graph result for both AI model at 75 epochs at 339 datasets and 3064 datasets respectively. Both the AI model achieved the desirable gradient for train/box loss and train/cls loss graph. However, when the researcher compares the val and metrics graphs, the AI model with 339 datasets has a lot of spikes whereas the AI model with 3064 datasets has a smoother line. This results the AI model with 339 datasets has an inconsistency when detecting hotspots causing some hotspots were not detected. Since that, the AI model with 3064 datasets has smoother line on the graphs, this results a more consistent results and more hotspots were detected by the AI model.

As a conclusion, the more datasets the AI model trains and tests, the better the overall performance of the AI model.

Table 4.9 Confusion Matrix at 75 epochs

Number of Epochs	Size of Datasets	True Positive (TP)	False Positive (FP)	True Negative (TN)	False Negative (FN)
75	339	0.67	0.33	1.00	0.00
75	3064	0.92	0.08	1.00	0.00

Table 4.10 Performance at 75 epochs

Number of Epochs	Size of Datasets	Accuracy	Precision	Recall	F1 Score
75	339	0.835	0.67	1.0	0.8024
75	3064	0.96	0.92	1.0	0.958

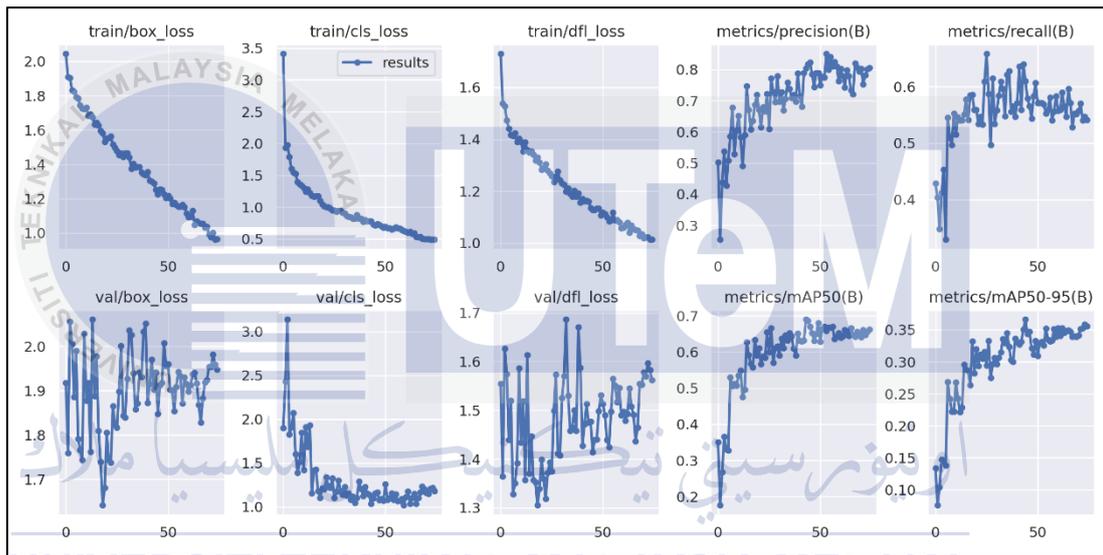


Figure 4.38 Results graph at 75 epochs with 339 datasets

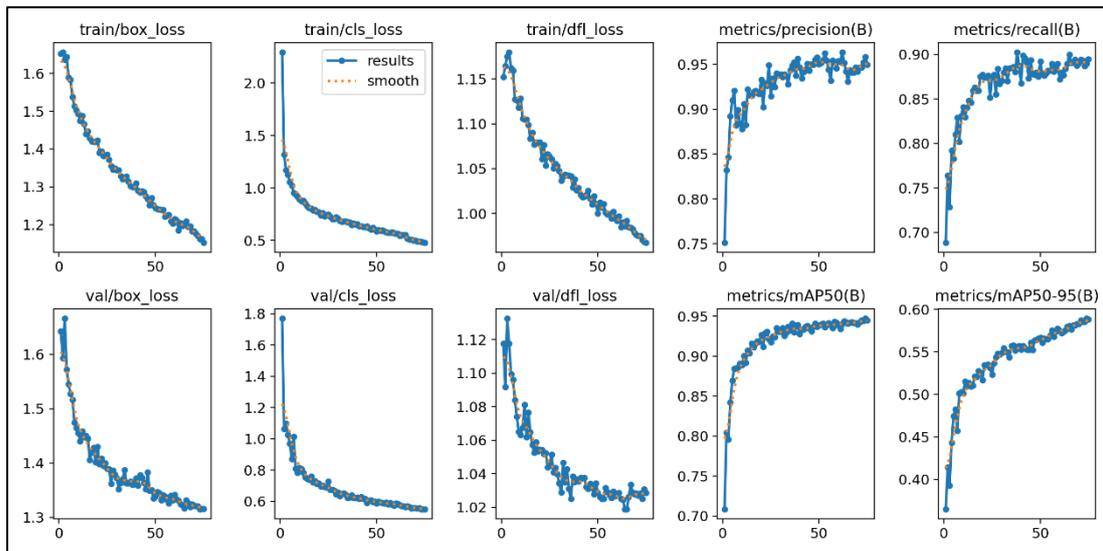


Figure 4.39 Results graph at 75 epochs with 3064 datasets

4.5 Graphical User Interface (GUI)

In this section, the researcher explains and discusses the graphical user interface (GUI) and its development for this research. A graphical user interface (GUI) is a digital interface that allows users to interact with graphical elements including icons, buttons, and menus. In a GUI, the visuals displayed in the user interface transmit pertinent information as well as actions that the user can perform. The GUI is designed and created for users to easily the AI model to detect hotspots on solar panels.

The researcher uses Visual Studio Code to code and create the GUI in which the researcher uses Python language to code. Before the researcher begins to code, the researcher must design the GUI not only to determine the elements that are needed but also to allow the researcher to properly visualize how the GUI to look like. The researcher uses a web-based tool called Make Real.Tldraw to design the GUI. Make Real.Tldraw allows the researcher to draw and design a user interface (UI) and make it real. However, the researcher only uses it to draw and design the GUI.

Figure 4.40 shows the researcher designs the GUI using Make Real.Tldraw. The blue boxes represent the buttons for GUI. On the top left, there are two buttons named “Choose Image” and Choose Button.” This allows the users to either choose only one image or a folder that contains many images for the AI model to detect hotspots. The light grey box on the left is where the original images will be displayed once the users have select. The right grey box is where the result images will be displayed once the AI model have generated the results. Below those light grey boxes are four buttons. These four buttons will allow the users to select between four AI models either 25, 50, 75 and 100 epochs. Below the four buttons is a box that will display the AI model directory path. Below the directory path box is the “Start” button that users will press once the users have selected the images and the AI model. At the top right shows the UTeM and Faculty of Electrical Engineering & Technology (FTKE) logo.

Once the researcher has a design for the GUI, the researcher begins making it into reality. The researcher codes the GUI using Python Language and Visual Studio Code (VSCode). The researcher installs Tkinter and CustomTkinter packages by using the pip command at the VSCode terminal. The Tkinter package is the standard Python interface to

the Tcl/Tk GUI toolkit. Tkinter are available on most Unix platforms, including macOS, as well as on Windows systems. Tkinter is not a thin wrapper, but adds a fair amount of its own logic to make the experience more pythonic. CustomTkinter is a python UI-library based on Tkinter, which provides new, modern and fully customizable widgets. CustomTkinter are created and used like normal Tkinter widgets and can also be used in combination with normal Tkinter elements. The widgets and the window colors either adapt to the system appearance or the manually set mode ('light', 'dark'), and all CustomTkinter widgets and windows support HighDPI scaling (Windows, macOS). With CustomTkinter the researcher gets a consistent and modern look across all desktop platforms (Windows, macOS, Linux).

Figure 4.41 shows the first stage of the GUI. This GUI only allows the user to choose an image and display it on the GUI. Although the GUI is very simple, this helps the researcher to understand and learn to code for choosing image in the system and display it on the GUI. Figure 4.42 shows the second stage of the GUI. The researcher not only now is able to display two images, the researcher is also able to choose the AI model and show the directory path for the AI model. The researcher simply defines the left and right frame for the GUI to display two images. However, the AI model is still not able to put into to use. Figure 4.43 shows the third stage of the GUI. The GUI now allows the AI model to be put into for practical application. The researcher defines the “Start” button with the code for prediction task for the GUI. The AI model detects whatever image that has been displayed on the left frame and generate result by detecting hotspots. Once the result has been generated, the result will be displayed on the right frame. The result is also saved in the PC. However, this GUI only allows the user to generate results for only one image.

Figure 4.44 shows the fourth stage of the GUI. The GUI now allows the users to select a folder that contains many images and generate result for every images. However, the users still need to browse through the PC to select the AI model. Figure 4.45 shows the fifth stage of the GUI. The GUI now allows the users to select between for AI model, 25, 50, 75 and 100 epochs, without browsing through the PC. The users can now simply click the buttons for 25, 50, 75 or 100 epochs. The researcher code the GUI to create four buttons and defines each button with the directory path for each AI model. However, the GUI still looks like in development and the researcher plans to make it look cleaner so that it will be user friendly and can be commercialize.

Figure 4.46 shows the final stage for the GUI. The GUI now looks cleaner, user friendly and can be commercialize. The researcher adjusts the position for each element of the GUI for the GUI to look cleaner and uniform. The researcher also adds the UTeM and FTKE logo. Figure 4.47 and 4.48 shows the popup message for the GUI. The GUI also display a popup message that tells the users where the results have been saved, the number of hotspots the AI model detected and the status of the solar panels whether there is a defect or not. This popup message will ease the users to instantly find the directory path for the results and know the number of hotspots detected and the status of the solar panels.

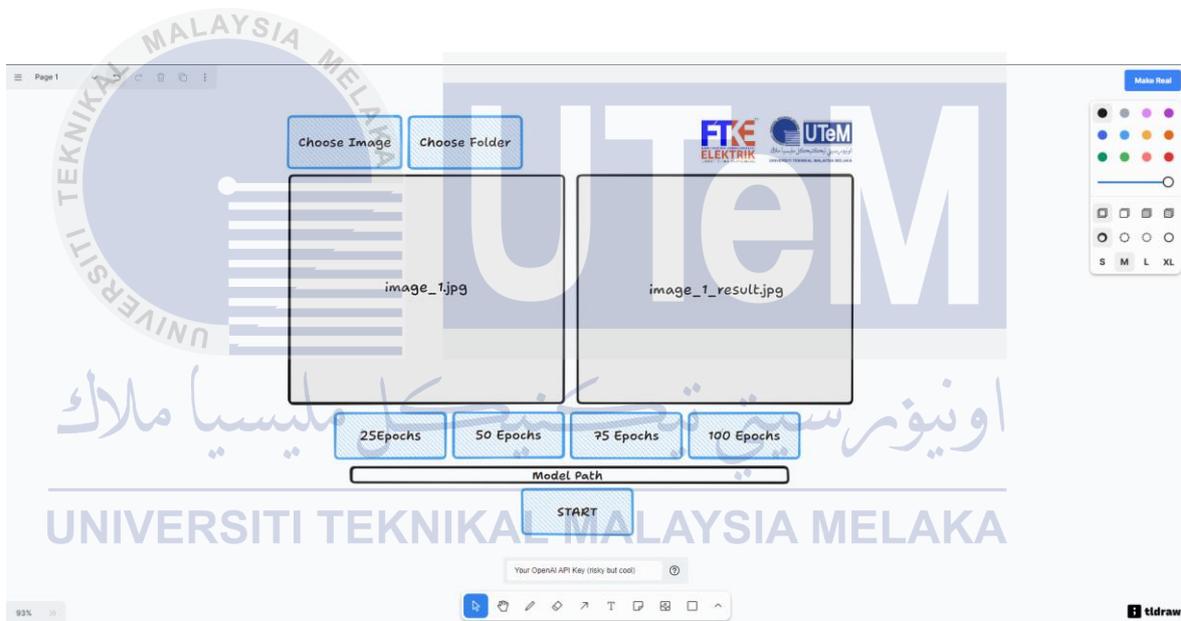


Figure 4.40 GUI design using Make Real.Tldraw

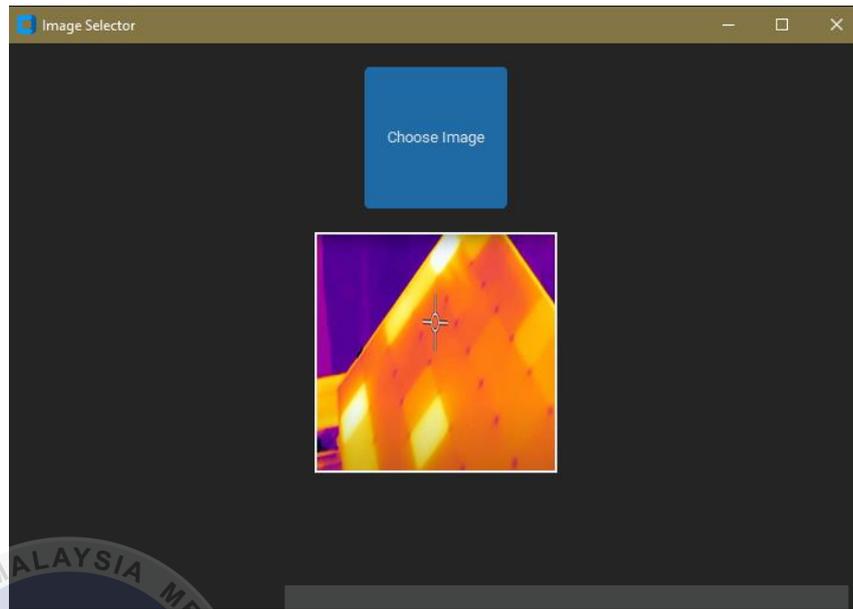


Figure 4.41 GUI first stage

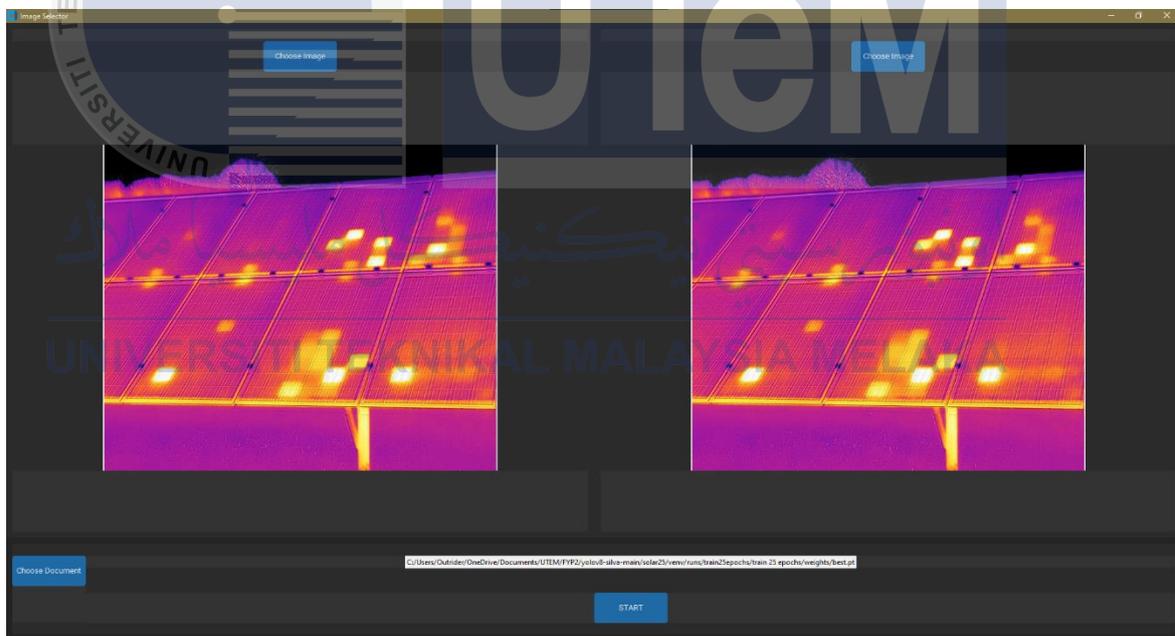


Figure 4.42 GUI second stage

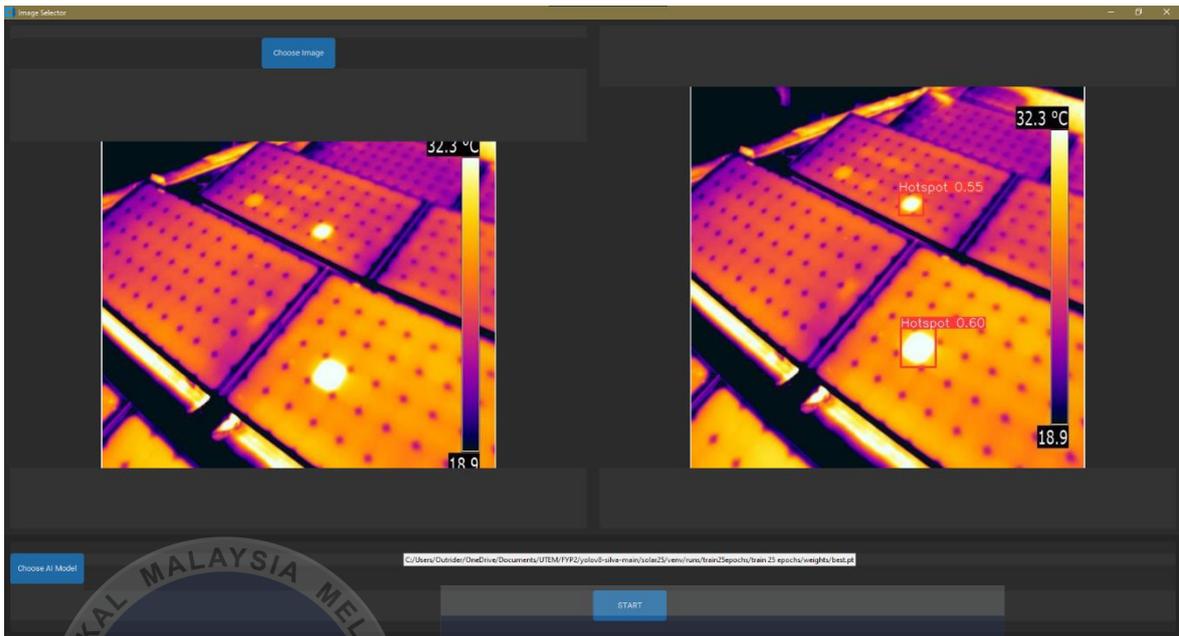


Figure 4.43 GUI third stage

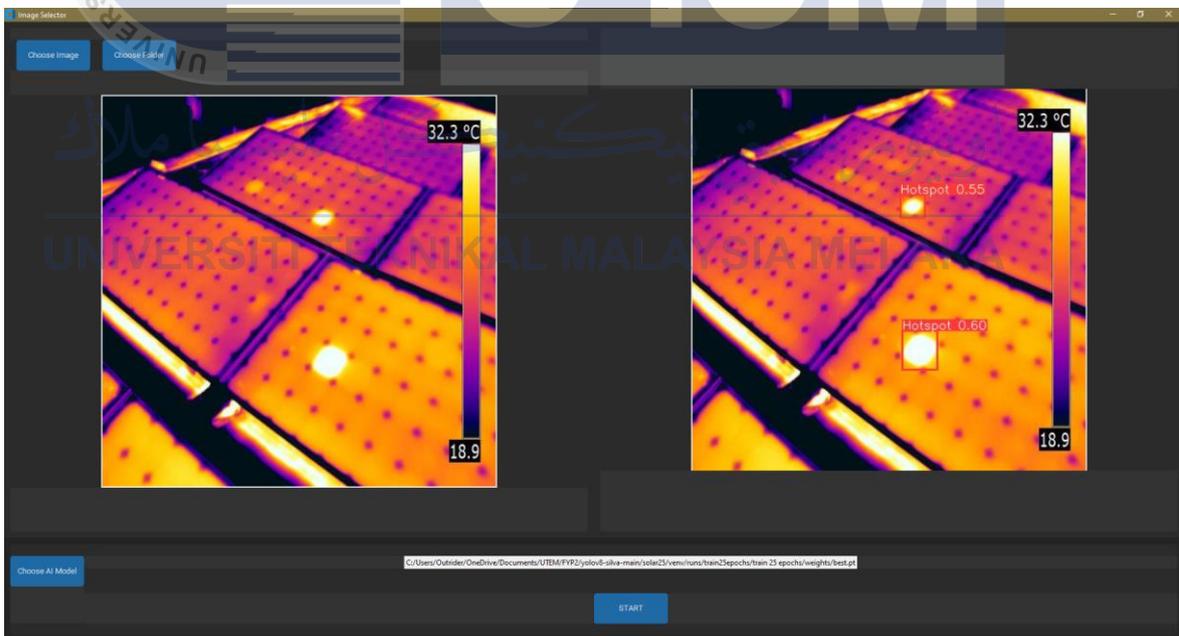


Figure 4.44 GUI fourth stage

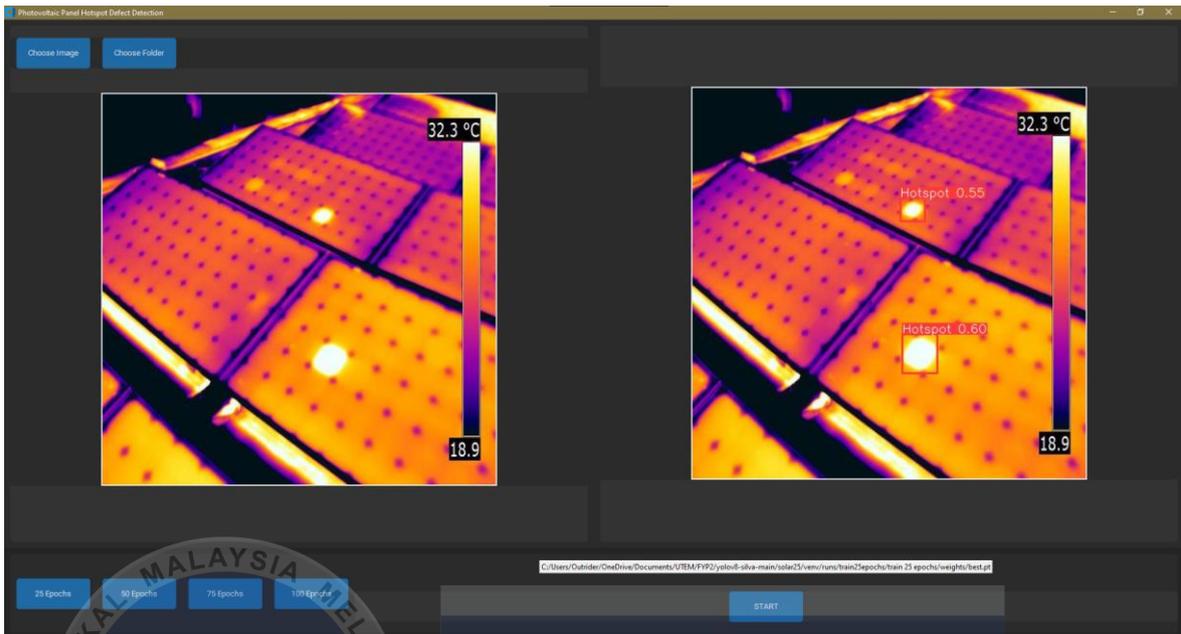


Figure 4.45 GUI fifth stage

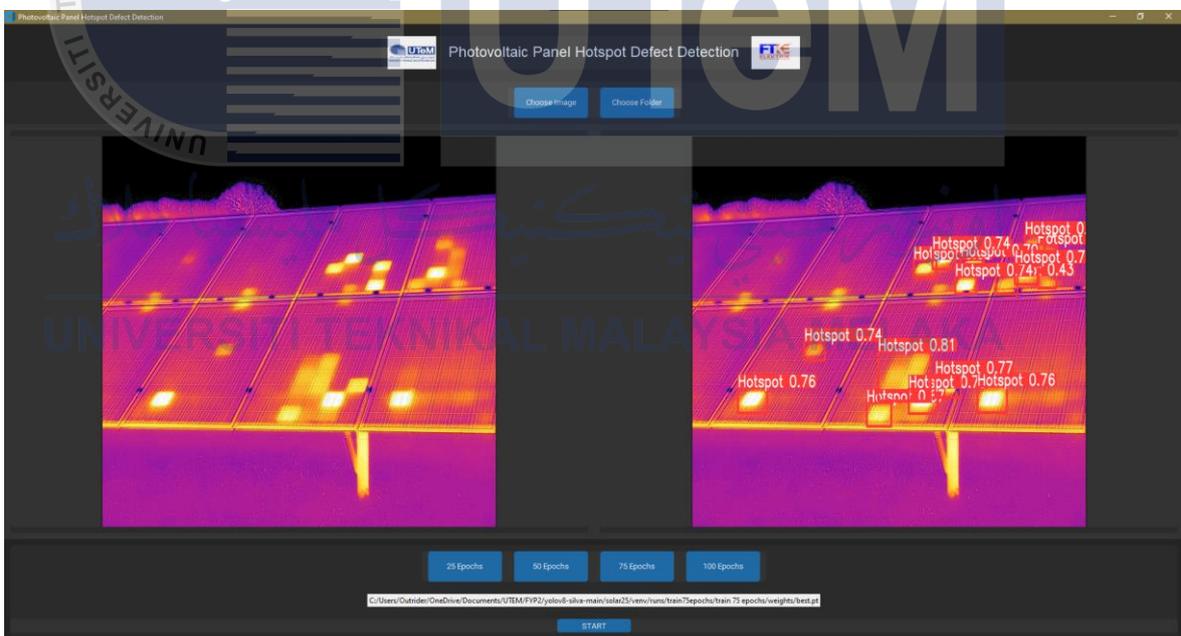


Figure 4.46 GUI final stage

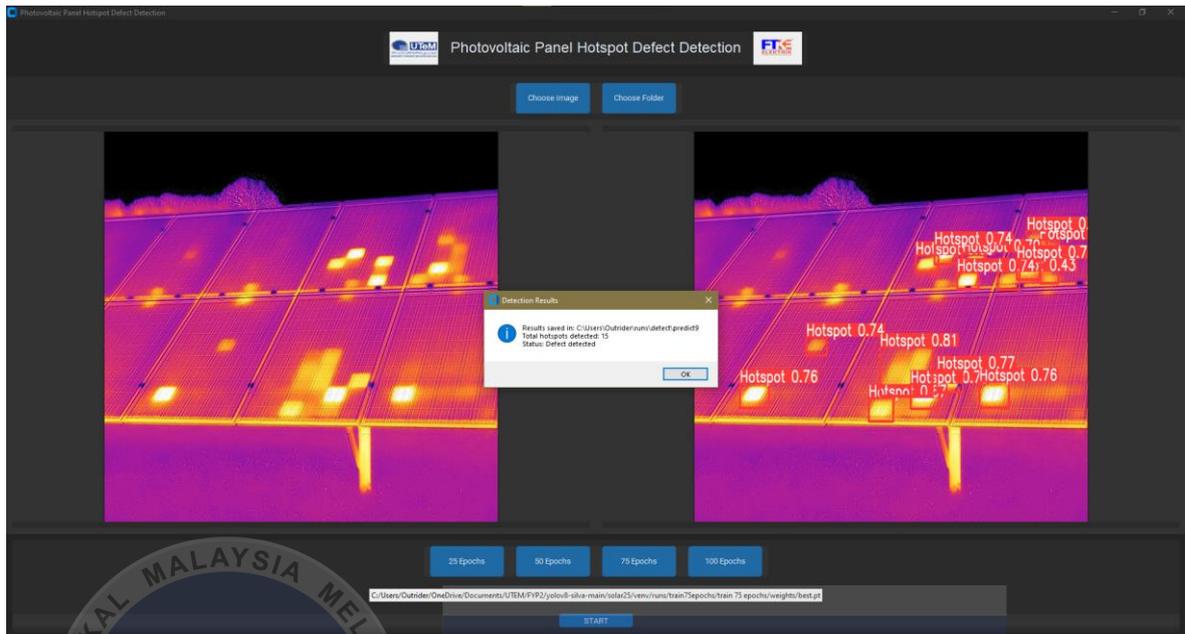


Figure 4.47 Popup message

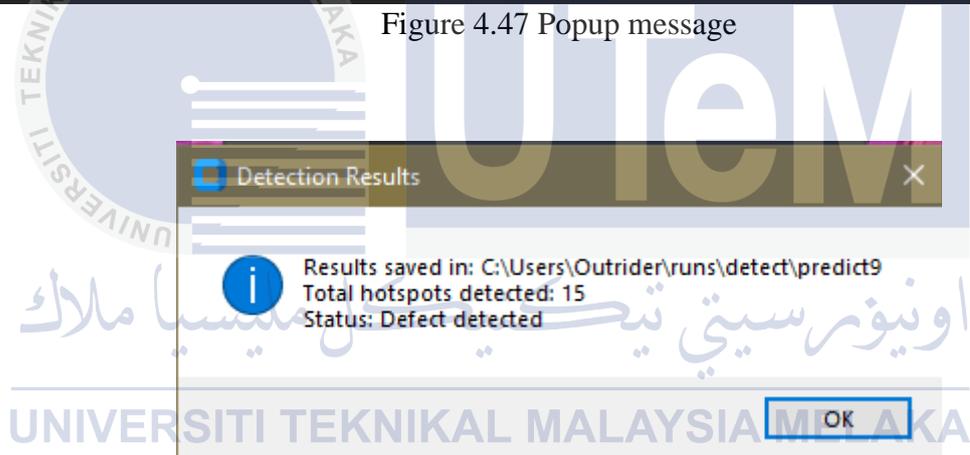


Figure 4.48 Popup message

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

Fossil fuels supply more than 80% of the energy in the world. However, the depletion of fossil fuels is inevitable. As a result, the demand for clean/environmentally friendly energy has greatly expanded recently. Renewable energy technologies are alluring energy sources that are clean, friendly to the environment, and help meet the world's energy needs. Renewable energy has multiple applications, including solar, wind, biomass, and geothermal. The most common application technology is solar energy, followed by wind energy. Solar energy is divided into two types: concentrated solar power (CSP) and solar photovoltaic (PV). The latter is more extensively used. Recent advancements in photovoltaic solar technology have led to widespread deployment of solar PV capacity to meet global energy demands.

However, solar panels will defect overtime. Defects such as hotspots, snail trails, delamination, cracks and microcracks will occur after some time. In this research, the researcher focusses on hotspots. When specific solar panel cells overheat due to localised shadowing, dirt buildup, or manufacturing defects, hotspots are created. These hotspots may reduce the overall output of the panel and harm damaged cells irreversibly. To locate hotspots, use thermal imaging during the day while the panels are exposed to direct sunlight. A hotspot is an area on the panel where the temperature is noticeably higher than it is elsewhere.

The researcher has come up with three objectives to overcome this problem and further boost the efficiency of solar panels. With this research, the researcher has achieved all three objectives. The first objective is to design a system that can easily detect defects on photovoltaic arrays of varying sizes and environmental conditions. The researcher design a system that uses a DJI Mavic 2 Enterprise drone that has a built-in high definition camera and thermal imaging device to fly autonomously above the photovoltaic panels using Drone

Harmony and capture thermal images and uses YOLO v8, an AI model, to detect hotspots on photovoltaic panels.

The second objective that has been achieved is to develop a model for AI-based defect detection system that refers to thermal imaging sensors. The researcher create a YOLO v8 model to detect hotspots on photovoltaic panels. The researcher uses Roboflow to annotate 339 images in FYP 1 and 3064 images in FYP 2 in order to help the YOLO v8 model to learn and recognize the hotspots. Then, the researcher splits the datasets into 70% for training, 20% for validation and 10% for testing. This method not only let the YOLO v8 model to learn the hotspots, but also test itself with unseen images of hotspots for practical application. Based on the results achieved by the researcher, it proves that the AI model is able to detect hotspots by using thermal imaging camera.

The third objective is to analyze the effectiveness and consistency of the AI-based defect detection system which uses YOLO v8. In both FYP, the researcher trains and tests the YOLO v8 model at 25, 50, 75 and 100 epochs and found that the YOLO v8 model at 75 epochs has the most favourable result. The researcher also found that there is an increase in the overall YOLO v8 performance when the researcher increases the size of datasets. This contradicts the researcher's hypothesis that the more extensive training the YOLO v8 model undergoes, the better the performance of the YOLO v8 model. However, the hypothesis that the more the size of datasets leads to better the performance of the YOLO v8 model remains valid. Thus, the researcher achieved the third objective of analyzing the effectiveness and consistency of the YOLO v8-based hotspot detection system.

As a conclusion, this research has proven to increase the efficiency of photovoltaic panels by using AI, drone and thermal imaging camera to detect hotspots. By implementing this system, it also decreases human labor. The researcher has personally installed 998 photovoltaic panels at Lotus Parit Raja, Batu Pahat. During the installation, a thermal inspection must be done. There were two people did the inspection and it took three days to complete. So, imagine if the researcher did the thermal inspection at a solar farm which contains thousands of photovoltaic panels. Not only that the inspection will consume a lot of time but the company needs to hire more people. This system also increases inspection efficiency. A human will feel tired but a machine don't. When humans feel tired, their efficiency in doing tasks decreases thus this affects the inspection efficiency. A machine

don't feel tired thus the inspection efficiency will not be affected. To conclude this research has definitely proven to increase the efficiency of photovoltaic panels by detecting hotspots.

5.2 Future Work

The researcher's future plans to improve the AI model's functionality and expand its use in photovoltaic (PV) panel defect detection are described in this part. These programs seek to improve PV system dependability and efficiency, which advances the overarching objective of maximizing the generation of solar energy.

Growing the number of the datasets used to train the AI model is one of the main goals for the future. A bigger dataset will include more sorts of defects, different climatic circumstances, and panel changes, which will enhance the model's ability to generalize. The AI model will be better able to recognize flaws in many scenarios by training on a more varied and larger dataset, which will improve performance and dependability. In order to complete this project, additional real-world data from different PV installations will need to be gathered and annotated. This may be accomplished by working with industry partners and making use of publicly accessible datasets.

Apart from augmenting the quantity of the dataset, the researcher intends to improve the AI model by integrating the identification of a broader range of imperfections. At the moment, common flaws like hotspots and shading problems are the main focus. Subsequent research endeavors aim to expand this potential to encompass more intricate and subtle flaws such as microcracks, delamination, cracks, and snail trails. If not found promptly, flaws and microcracks—which are frequently undetectable to the unaided eye—can seriously impair panel efficiency. PV panel delamination, or the separation of panel layers, can result in moisture intrusion and additional damage. Discoloration lines, or snail trails, can reveal underlying chemical reactions and moisture problems. The AI model will be trained to identify these extra flaws, giving the system a more complete diagnostic tool for preserving the health of PV panels.

In order to increase functionality and usability, the researcher also intends to improve the user interface. The redesigned interface will let users prioritize whatever type of defect detection to utilize instead of forcing them to select the number of training epochs. Thanks

to this customization, customers will be able to adjust the AI model's focus to suit their own operating requirements or common problems with their PV systems. With an interface like this, the tool will be easier to use and more flexible to meet the needs of many users, which will increase its usefulness.

One important improvement for future efforts is to include localization features in the output of the AI model. Maintaining a PV array will be made much easier with the capacity to locate flaws precisely. The time and effort required for inspections will be decreased since users will be able to easily locate and access the precise panels that need maintenance. By combining GPS data with the inspection drones and using sophisticated image processing techniques, this localization capability will be accomplished. The solution will improve overall operating efficiency and ease the maintenance process by giving exact position information.

Lastly, the researcher intends to enhance the AI model's capacity to automatically classify inspection results into panels with and without problems in order to further expedite the defect identification procedure. Users will be able to prioritize maintenance tasks by immediately determining the amount of problems within their PV installations thanks to this automated categorization. The technology will facilitate more effective resource allocation and early interventions by providing a comprehensive overview of the defect status across all inspected panels.

In conclusion, the work described in this part will hopefully greatly improve the AI model's ability to identify and maintain PV panel defects. The researcher aims to develop a robust and comprehensive tool by increasing the size of the dataset, diversifying the types of problems that may be detected, enhancing the user interface, incorporating automatic classification and localization features, incorporating predictive maintenance, and guaranteeing scalability. These developments will serve the larger objective of renewable energy adoption and sustainability by optimizing solar energy systems.

REFERENCES

- [1] H. H. Pourasl, R. V. Barenji, and V. M. Khojastehnezhad, "Solar energy status in the world: A comprehensive review," *Energy Reports*, vol. 10. Elsevier Ltd, pp. 3474–3493, Nov. 01, 2023. doi: 10.1016/j.egy.2023.10.022.
- [2] M. T. Hussain and E. J. Mahdi, "Assessment of Solar Photovoltaic Potential in Iraq," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jun. 2018. doi: 10.1088/1742-6596/1032/1/012007.
- [3] "Photovoltaics – Department of Energy".
- [4] G. Terzoglou, M. Loufakis, P. Symeonidis, D. Ioannidis, and D. Tzovaras, "Employing deep learning framework for improving solar panel defects using drone imagery," in *International Conference on Digital Signal Processing, DSP*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/DSP58604.2023.10167960.
- [5] "Common Solar Panel Defects".
- [6] "How Long Do Solar Panels Last".
- [7] E. Irmak, S. Melike, S. G. Ayaz, A. Gok, and S. Busra, "A Survey on Public Awareness Towards Renewable Energy in Turkey."
- [8] L. Elie, C. Granier, and S. Rigot, "The different types of renewable energy finance: A Bibliometric analysis," *Energy Econ*, vol. 93, Jan. 2021, doi: 10.1016/j.eneco.2020.104997.
- [9] E. Brudler *et al.*, "24 years Postgraduate Program Renewable Energy." [Online]. Available: <http://www.zeva.org/>
- [10] S. Singer, J.-P. Denruyter, and D. Yener, "The Energy Report: 100 % Renewable Energy by 2050," 2017, pp. 379–383. doi: 10.1007/978-3-319-45659-1_40.
- [11] IPCC, "The evidence is clear: the time for action is now. We can halve emissions by 2030," IPCC.
- [12] J. Blazquez, R. Fuentes-Bracamontes, C. A. Bollino, and N. Nezamuddin, "The renewable energy policy Paradox," *Renewable and Sustainable Energy Reviews*, vol. 82, pp. 1–5, Feb. 2018, doi: 10.1016/j.rser.2017.09.002.
- [13] B. P. Heard, B. W. Brook, T. M. L. Wigley, and C. J. A. Bradshaw, "Burden of proof: A comprehensive review of the feasibility of 100% renewable-electricity

- systems,” *Renewable and Sustainable Energy Reviews*, vol. 76, pp. 1122–1133, Sep. 2017, doi: 10.1016/j.rser.2017.03.114.
- [14] A. Harjanne and J. M. Korhonen, “Abandoning the concept of renewable energy,” *Energy Policy*, vol. 127, pp. 330–340, Apr. 2019, doi: 10.1016/j.enpol.2018.12.029.
- [15] S. A. A. Shah and Y. A. Solangi, “A sustainable solution for electricity crisis in Pakistan: opportunities, barriers, and policy implications for 100% renewable energy,” *Environmental Science and Pollution Research*, vol. 26, no. 29, pp. 29687–29703, Oct. 2019, doi: 10.1007/s11356-019-06102-0.
- [16] W. Zappa, M. Junginger, and M. van den Broek, “Is a 100% renewable European power system feasible by 2050?,” *Appl Energy*, vol. 233–234, pp. 1027–1050, Jan. 2019, doi: 10.1016/j.apenergy.2018.08.109.
- [17] A. Aghahosseini, D. Bogdanov, and C. Breyer, “Towards sustainable development in the MENA region: Analysing the feasibility of a 100% renewable electricity system in 2030,” *Energy Strategy Reviews*, vol. 28, p. 100466, Mar. 2020, doi: 10.1016/j.esr.2020.100466.
- [18] D. Bogdanov, A. Gulagi, M. Fasihi, and C. Breyer, “Full energy sector transition towards 100% renewable energy supply: Integrating power, heat, transport and industry sectors including desalination,” *Appl Energy*, vol. 283, p. 116273, Feb. 2021, doi: 10.1016/j.apenergy.2020.116273.
- [19] A. S. Oyewo *et al.*, “Just transition towards defossilised energy systems for developing economies: A case study of Ethiopia,” *Renew Energy*, vol. 176, pp. 346–365, Oct. 2021, doi: 10.1016/j.renene.2021.05.029.
- [20] A. Gulagi, M. Alcanzare, D. Bogdanov, E. Esparcia, J. Ocon, and C. Breyer, “Transition pathway towards 100% renewable energy across the sectors of power, heat, transport, and desalination for the Philippines,” *Renewable and Sustainable Energy Reviews*, vol. 144, p. 110934, Jul. 2021, doi: 10.1016/j.rser.2021.110934.
- [21] D. Gielen, F. Boshell, D. Saygin, M. D. Bazilian, N. Wagner, and R. Gorini, “The role of renewable energy in the global energy transformation,” *Energy Strategy Reviews*, vol. 24, pp. 38–50, Apr. 2019, doi: 10.1016/j.esr.2019.01.006.
- [22] L. Cozzi, T. Gould, S. Bouckart, D. Crow, T. Kim, and C. Mcglade, *World Energy Outlook 2020*,. Paris: International Energy Agency, 2020.

- [23] D. Raimi, C. Erin, G.N. Richard, P. Brian, V. Seth, and W. Jordan, *Global Energy Outlook 2022: Turning Points and Tension in the Energy Transition*. Washington, DC: Resources for the Future, 2022.
- [24] P. A. Owusu and S. Asumadu-Sarkodie, “A review of renewable energy sources, sustainability issues and climate change mitigation,” *Cogent Eng*, vol. 3, no. 1, p. 1167990, Dec. 2016, doi: 10.1080/23311916.2016.1167990.
- [25] N. L. Panwar, S. C. Kaushik, and S. Kothari, “Role of renewable energy sources in environmental protection: A review,” *Renewable and Sustainable Energy Reviews*, vol. 15, no. 3, pp. 1513–1524, Apr. 2011, doi: 10.1016/j.rser.2010.11.037.
- [26] A. Qazi *et al.*, “Towards Sustainable Energy: A Systematic Review of Renewable Energy Sources, Technologies, and Public Opinions,” *IEEE Access*, vol. 7, pp. 63837–63851, 2019, doi: 10.1109/ACCESS.2019.2906402.
- [27] IRENA, “Solar Energy,” International Renewable Energy Agency. www.irena.org/solar.
- [28] A. O. M. Maka and J. M. Alabid, “Solar energy technology and its roles in sustainable development,” *Clean Energy*, vol. 6, no. 3, pp. 476–483, Jun. 2022, doi: 10.1093/ce/zkac023.
- [29] M. S. Răboacă *et al.*, “Concentrating Solar Power Technologies,” *Energies (Basel)*, vol. 12, no. 6, p. 1048, Mar. 2019, doi: 10.3390/en12061048.
- [30] Keith Lovegrove and Wes Stein, *Concentrating Solar Power Technology: Principles, Developments and Applications*. Woodhead Publishing, 2012.
- [31] H. Müller-Steinhagen and F. Trieb, “Concentrating solar power, - A review of the technology,” *Ingenia*, vol. 18, pp. 43–50, Jan. 2004.
- [32] W. Priharti, A. F. K. Rosmawati, and I. P. D. Wibawa, “IoT based photovoltaic monitoring system application,” *J Phys Conf Ser*, vol. 1367, no. 1, p. 012069, Nov. 2019, doi: 10.1088/1742-6596/1367/1/012069.
- [33] M. Ş. Kalay, B. Kılıç, and Ş. Sağlam, “Systematic review of the data acquisition and monitoring systems of photovoltaic panels and arrays,” *Solar Energy*, vol. 244, pp. 47–64, Sep. 2022, doi: 10.1016/j.solener.2022.08.029.
- [34] D. Gielen, R. Gorini, N. Wagner, R. Leme, L. Gutierrez, and G. Prakash, “Global energy transformation: a roadmap to 2050,” IRENA.

- [35] G. Badran and M. Dhimish, "Potential Induced Degradation in Photovoltaic Modules: A Review of the Latest Research and Developments," *Solar*, vol. 3, no. 2, pp. 322–346, Jun. 2023, doi: 10.3390/solar3020019.
- [36] S. P. Pathak, D. S. Patil, and S. Patel, "Solar panel hotspot localization and fault classification using deep learning approach," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 698–705. doi: 10.1016/j.procs.2022.08.084.
- [37] E. H. Sepúlveda Oviedo, L. Travé-Massuyès, A. Subias, C. Alonso, and M. Pavlov, "Feature extraction and health status prediction in PV systems," *Advanced Engineering Informatics*, vol. 53, Aug. 2022, doi: 10.1016/j.aei.2022.101696.
- [38] Z. A. Jaffery, A. K. Dubey, Irshad, and A. Haque, "Scheme for predictive fault diagnosis in photo-voltaic modules using thermal imaging," *Infrared Phys Technol*, vol. 83, pp. 182–187, Jun. 2017, doi: 10.1016/j.infrared.2017.04.015.
- [39] C. Dunderdale, W. Brettenny, C. Clohessy, and E. Van Dyk, "Photovoltaic defect classification through thermal infrared imaging using a machine learning approach," *Progress in Photovoltaics: Research and Applications*, vol. 28, Dec. 2019, doi: 10.1002/pip.3191.
- [40] H. Munawer Al-Otum, "Deep learning-based automated defect classification in Electroluminescence images of solar panels," *Advanced Engineering Informatics*, vol. 58, Oct. 2023, doi: 10.1016/j.aei.2023.102147.
- [41] B. Sandeep, D. S. Reddy, R. Aswin, and R. Mahalakshmi, "Monitoring of PV Modules and Hotspot Detection using TensorFlow," in *Proceedings of the International Conference on Electronics and Renewable Systems, ICEARS 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 155–160. doi: 10.1109/ICEARS53579.2022.9752346.
- [42] J. Huang, K. Zeng, Z. Zhang, and W. Zhong, "Solar panel defect detection design based on YOLO v5 algorithm," *Heliyon*, vol. 9, no. 8, Aug. 2023, doi: 10.1016/j.heliyon.2023.e18826.
- [43] A. Saranya and R. Subhashini, "A systematic review of Explainable Artificial Intelligence models and applications: Recent developments and future trends," *Decision Analytics Journal*, vol. 7. Elsevier Inc., Jun. 01, 2023. doi: 10.1016/j.dajour.2023.100230.
- [44] "YOLO v8! The real state-of-the-art__ by Chinmay Bhalerao _ MLearning.ai _ Medium".

- [45] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, “A Review of Yolo Algorithm Developments,” in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 1066–1073. doi: 10.1016/j.procs.2022.01.135.
- [46] M. Uma, S. Abirami, M. Ambika, M. Kavitha, S. Sureshkumar, and K. R., “A Review on Augmented Reality and YOLO,” in *2023 4th International Conference on Smart Electronics and Communication (ICOSEC)*, IEEE, Sep. 2023, pp. 1025–1030. doi: 10.1109/ICOSEC58147.2023.10275842.
- [47] “YOLOv8 vs. YOLOv5_ Choosing the Best Object Detection Model”.
- [48] A. N. Wilson, K. A. Gupta, B. H. Koduru, A. Kumar, A. Jha, and L. R. Cenkeramaddi, “Recent Advances in Thermal Imaging and its Applications Using Machine Learning: A Review,” *IEEE Sens J*, vol. 23, no. 4, pp. 3395–3407, Feb. 2023, doi: 10.1109/JSEN.2023.3234335.
- [49] M. Kurosaki, D. Kawaguchi, K. Ogawa, R. Nakamura, and H. Hadama, “Comparison of Drone Classification Accuracy for CNN Models Using UWB Radar,” in *2022 Asia-Pacific Microwave Conference (APMC)*, 2022, pp. 907–909. doi: 10.23919/APMC55665.2022.10000022.
- [50] N. V. Sailaja *et al.*, “Drone Automation: An Overview of Recent Progress and Research,” in *Proceedings - 2022 4th International Conference on Advances in Computing, Communication Control and Networking, ICAC3N 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 1335–1339. doi: 10.1109/ICAC3N56670.2022.10074234.