PERFORMANCE ANALYSIS OF CONVOLUTIONAL NEURAL NETWORKS FOR RECOGNITION TASK OF SIMILAR INDUSTRIAL MACHINING PARTS

MUHAMMAD FIKRI BIN FAHAR

BACHELOR OF MANUFACTURING ENGINEERING TECHNOLOGY WITH HONOURS

2022

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Faculty of Mechanical and Manufacturing Engineering Technology

PERFORMANCE ANALYSIS OF CONVOLUTIONAL NEURAL NETWORKS FOR RECOGNITION TASK OF SIMILAR INDUSTRIAL MACHINING PARTS

Muhammad Fikri Bin Fahar

Bachelor of Manufacturing Engineering Technology with Honours

2022

PERFORMANCE ANALYSIS OF CONVOLUTIONAL NEURAL NETWORKS FOR RECOGNITION TASK OF SIMILAR INDUSTRIAL MACHINING PARTS

MUHAMMAD FIKRI BIN FAHAR

A thesis submitted
in fulfillment of the requirements for the degree of
Bachelor of Manufacturing Engineering Technology with Honours

Faculty of Mechanical and Manufacturing Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2022

DECLARATION

I declare that this Choose an item. entitled “Performance Analysis of Convolutional Neural Networks for Recognition Task of Similar Industrial Machining Parts” is the result of my own research except as cited in the references. The Choose an item. has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	FIKRI
Name	:	MUHAMMAD FIKRI BIN FAHAR
Date	:	18/1/2022

APPROVAL

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the Bachelor of Manufacturing Engineering Technology with Honours.

Signature : Hadyan Hafizh

Dr. Hadyan Hafizh
Pensyarah Kanan
Jabatan Teknologi Kejuruteraan Pembuatan
Fakulti Teknologi Kejuruteraan Mekanikal dan Pembuatan

Supervisor Name : DR HADYAN HAFIZH

Date : 18/1/2022

DEDICATION

This work is humbly dedicated to all my valuable treasures:

For my beloved parents:

Fahar Bin Ruslan

Sidah Binti Yusuf

For my supportive siblings:

Adi Bin Fahar

Kamariah Binti Fahar

Nurazikin Binti Fahar

Nurfa Binti Fahar

Thank you for love, sacrifices and always there in my every step of life.

For my respected supervisor:

Dr. Hadyan Hafizh

For all UTeM lecturers and my treasured friends.

Who gave me strength and faith to overcome all the difficulties.

ABSTRACT

With recent advances in convolutional neural network (CNN), there is an increased interest in applying this technology to industrial image processing. Specifically for automatic recognition of similar industrial machining parts. Misclassifying parts is one of the major issues faced by the operator in small and medium enterprises (SMEs) manufacturing industry. The manual inspection currently employed leaves room for human error due to inability to distinguish similar parts. As a result, incorrect machining parts could be sent to the costumer, decreasing the company’s reputation in the eye of their existing and potential customers. To overcome the traceability issue of similar machining parts, it is critical to incorporate automated and digital inspection. Therefore, the current work investigates the performance of different CNN models that can be integrated into a machine-vision system to perform automatic recognition tasks.

ABSTRAK

Dengan kemajuan terkini dalam rangkaian neural konvolusi (CNN), terdapat peningkatan minat untuk menggunakan teknologi ini pada pemprosesan imej industri. Khusus untuk pengecaman automatik bahagian pemesinan industri yang serupa. Salah klasifikasi bahagian adalah salah satu isu utama yang dihadapi oleh pengendali dalam industri pembuatan perusahaan kecil dan sederhana (SME). Pemeriksaan manual yang digunakan pada masa ini meninggalkan ruang untuk kesilapan manusia kerana ketidakupayaan untuk membezakan bahagian yang serupa. Akibatnya, bahagian pemesinan yang salah boleh dihantar kepada pelanggan, mengurangkan reputasi syarikat di mata pelanggan sedia ada dan bakal pelanggan mereka. Untuk mengatasi isu kebolehkesanan bahagian pemesinan yang serupa, adalah penting untuk menggabungkan pemeriksaan automatik dan digital. Oleh itu, kerja semasa menyiasat prestasi model CNN berbeza yang boleh disepadukan ke dalam sistem penglihatan mesin untuk melaksanakan tugas pengecaman automatik.

ACKNOWLEDGEMENTS

In the Name of Allah, the Most Gracious, the Most Merciful

First and foremost, I would like to thank and praise Allah the Almighty, my Creator, my Sustainer, for everything I received since the beginning of my life. I would like to extend my appreciation to the Universiti Teknikal Malaysia Melaka (UTeM) for providing the research platform.

My utmost appreciation goes to my main supervisor, Dr. Hadyan Hafizh, Faculty of Manufacturing and Mechanical Engineering Technology, Universiti Teknikal Malaysia Melaka for all his support, advice, and inspiration. His constant patience for guiding and providing priceless insights will forever be remembered. Also, my teammates PSM thanks go to Mr. Ahmad Bukhari Bin Abd Halim, Mr. Muhammad Fidaus Bin Idris and Mr. Muhammad Nazmeer Bin Ali for all the help and support for this project and also my small family BMMW.

Last but not least I am most grateful to my parents for their endless love and prayers throughout my life. I did not forget all of their sacrifices to provide me with a comfortable life and a good education. Without them, I will not be where I am today. Apart from them, I would like to thank my whole family for always taking an interest in the things that I work on, giving me strength and support during difficult times Finally, thank you to all the individual(s) who had provided me the assistance, support and inspiration to embark on my study.

TABLE OF CONTENTS

		PAGE
DECLARATION		
APPROVAL		
DEDICATION		
ABSTRACT		**i**
ABSTRAK		**ii**
ACKNOWLEDGEMENTS		**iii**
TABLE OF CONTENTS		**iv**
LIST OF TABLES		**vi**
LIST OF FIGURES		**vii**
LIST OF APPENDICES		**ix**
CHAPTER 1	**INTRODUCTION**	**1**
1.1	Research Background and Motivation	1
1.2	Problem Statement	4
1.3	Research Question	5
1.4	Research Objective	5
1.5	Research Scope and Limitation	6
1.6	Research Design and Planning	7
	1.6.1 Research Framework	7
	1.6.2 Gantt Chart and Milestones	8
CHAPTER 2	**LITERATURE REVIEW**	**2**
2.1	Introduction	2
2.2	Deep Transfer Learning	2
	2.2.1 Deep Learning	2
	2.2.2 Transfer Learning	3
2.3	Convolutional Neural Networks (CNN)	4
	2.3.1 CNN Mechanism	4
	2.3.2 Transfer Learning CNN Models	6
2.4	Summary	7
CHAPTER 3	**METHODOLOGY**	**9**
3.1	Introduction	9
3.1	Research Methodology	10
3.1	Design of Experiment	11
	3.1.1 Data Collection	12

	3.1.2	Model Training	14
	3.1.3	Performance Analysis	14
3.2	Summary		15
CHAPTER 4	**RESULTS AND DISCUSSION**		**17**
4.1	Introduction		17
4.2	Data Collection		17
	4.2.1	True Label Data Collection	17
	4.2.2	Data Augmentation	18
4.3	Model Training		20
	4.3.1	Case 1: MobileNet	20
	4.3.2	NasNetMobile	22
4.4	Performance Analysis		23
	4.4.1	Test Results	23
	4.4.2	Confusion Matrix	28
4.5	Summary		36
CHAPTER 5	**CONCLUSION AND RECOMMENDATIONS**		**37**
5.1	Conclusion		37
5.2	Recommendations		38
5.3	Project Potential		38
REFERENCES			**39**
APPENDICES			**48**

LIST OF TABLES

TABLE	TITLE	PAGE
Table 1	Gantt chart of PSM 2	8
Table 2	Procedure works	19
Table 3	MobileNet Performance Metrices	25
Table 4	NasNetNetMobile Performance	27
Table 5	Performance Metrices of CNN MobileNet model	29
Table 6	Performance metrics of CNN MobileNet model	30

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 1 : Research Framework		7
Figure 2 Architecture of the CNN model		5
Figure 3 : Proposed research methodology		10
Figure 4 : Design of experiment		11
Figure 5 Sample of original images		12
Figure 6 Data structure employed in the current work		13
Figure 7Architecture of the CNN model		18
Figure 8 Imported libraries		20
Figure 9 Defined data directory		21
Figure 10 Class label		21
Figure 11 Base model		22
Figure 12 Data saving		22
Figure 13 Model name changes in libraries		23
Figure 14 Base model name change		23
Figure 15 Data model name change		23
Figure 16 Epoch 1 Train accuracy for MobileNet		31
Figure 17 Epoch 2 Train accuracy for MobileNet		31
Figure 18 Epoch 3 Train accuracy for MobileNet		32
Figure 19 Epoch 1 Test accuracy for MobileNet Model		32
Figure 20 Epoch 2 Test accuracy for MobileNet Model		33
Figure 21 Epoch 3 Test accuracy for MobileNet Model		33

Figure 22 Epoch 1 Train accuracy for NasNet 34

Figure 23 Epoch 2 Train accuracy for NasNet 34

Figure 24 Epoch 3 Train accuracy for NasNet 34

Figure 25 Epoch 1 Test accuracy for NasNet 35

Figure 26 Epoch 2 Test accuracy for NasNet 35

Figure 27 Epoch 3 Test accuracy for NasNet 35

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
APPENDIX A	List of distribution Performance Metrices	48

CHAPTER 1

INTRODUCTION

1.1 Research Background and Motivation

Modern industrial firms are increasingly combining computer vision and automation into their processes in order to achieve higher quality and more accurate inspection of things. Deep learning, for example, is a sort of artificial intelligence-based technology that supports the business in automating processes that need minimal human intervention or supervision. In neural network applications, the convolutional neural network (CNN) is one of the most often utilised forms of neural networks. Deep learning is being used to develop important approaches for computer vision. CNN is employed in a broad range of different applications nowadays. The classification of images and the detection of defects in industrial products are two of the most common applications of CNN. In recent research CNN was utilised to investigate the detection of flaws in aluminium alloys during robotic arc welding. To enhance the CNN dataset in this work, data augmentation and noise addition were used. The CNN model was found to be 99.38 percent accurate. (Zhang et al., 2019). An investigation on the use of CNN to identify abnormalities in selective laser sintering (SLS) was also conducted in another study. When utilised to categorise good and faulty photos during the manufacture of components, two transfer learning CNN models with pre-trained weights were shown to be effective in classifying both excellent and defective images. According to the findings, the VGG16 transfer learning CNN model generated the greatest results, with an accuracy of 95.8 percent on the testing procedure. (Westphal & Seitz, 2021). Researchers discovered that the CNN model can achieve the highest accuracy rates of 71 percent, 89 percent, and 95 percent for weldment classifications that are divided into six categories. The CNN model also achieved the highest accuracy rates of 71 percent, 89 percent, and 95 percent for weldment classifications that are divided into four categories.(Bacioiu et al., 2019a). Another research, which employed the SS304 TIG welding method, had come to the conclusion that CNN was capable of learning strong representations of welding faults, which had previously been published.(Bacioiu et al., 2019b). Using CNN, it was possible to classify welds into two categories: good and defective. This accuracy was reported to be approximately 89.5 percent.

The use of CNN for the examination of metal additive manufacturing components has been examined in further depth in this study. In order to avoid overfitting difficulties, the regularisation and dropout layers were added to the CNN architecture for the objectives of this research. The accuracy of the CNN model has been improved by around 92.1 percent as a result of data augmentation strategies.(Cui et al., 2020). A similar investigation, in which CNN was used to identify faults in metallic surfaces, was also conducted out and the findings were published. In addition, data augmentation was performed to increase the overall quality of the training data set. Furthermore, the study led to the establishment of a new dataset for the identification of metallic surface defects, which was named the GC10-DET dataset. The suggested technique for the identification of metallic flaws, which made use of this dataset and the CNN model, was found to meet the accuracy standards for the detection of metallic defects when tested.(Lv et al., 2020). The question of whether CNN might be used to metal production components was also studied. In this work, four CNN models were employed to identify weld faults in galvanised steel sheets, and the findings were reported in a peer-reviewed publication. It was discovered, as a result of this research, that the VGG16 transfer learning CNN model paired with the data augmentation approach was proven to be the most effective model for obtaining state-of-the-art performance in the detection of weld faults.(Ma et al., 2021)

The usage of CNN is becoming more popular in a range of industrial domains because to its better classification skills, notably in the identification of metal and welding faults. In the actual world, CNN has shown to be very successful at completing recognition and classification tasks.(Deshpande et al., 2020; He et al., 2021; Miao et al., 2021; Yang et al., 2020). CNN has also been used to discover and study problems in the casting process, in addition to its usage in casting applications. In order to increase the performance of the CNN model, researchers performed a study in which they employed synthetic faults to make it more accurate and reliable. An architecture for convolutional neural networks (CNNs) called Xnet-II was designed in this paper and utilised in the study. It comprises 30 layers and more than 1,350,000 parameters.(Mery, 2020). X-ray pictures of casting products were employed as inputs to the CNN model in another investigation, which resulted in an improved prediction. It has been stated that the accuracy of the tests performed may go up to 95.5 percent in certain cases.(Jiang et al., 2021). It was determined whether or not there were any faults in the casting product (such as blowholes, chipping, cracks, and automated washing), as well as other aspects. The researchers sent 6000 photos with a resolution of 768 × 768 pixels into the CNN model, which produced an accurate result. According to the researchers, the training model obtained an experimental accuracy of more than 98 percent using real-world experiments.(Nguyen et al., 2021).

The CNN model has been used to identify faults in industrial items for quite some time, and earlier researchers have made substantial efforts in this area,(Ji et al., 2021; Nagata et al., 2019; Pan et al., 2020; Perera et al., 2021; Snow et al., 2021; Staar et al., 2019) there is still more work to be done. There has been little attention paid to the recognition of similar industrial machining parts. It is not only defects that are a source of concern for human operators on the manufacturing floor, but also the misclassification of machining components that they must contend with.

As a consequence of the resemblance between two machining components, this issue emerges, and when handled by a human operator, it presents the possibility of human mistake. When it comes to misclassification of machined components on the plant floor, it is a serious issue that happens regularly. It has been discovered that customers have gotten erroneous components owing to a lack of current traceability, and the same situation is recurring again and again. It has a negative impact on the company's reputation among consumers and suppliers. It has a negative impact on the company's reputation among current customers and vendors. It is critical that manual inspections be replaced as soon as feasible by digital and automated examinations. Because of this, the present study suggests that a CNN model be used to recognise and categorise two machining components that are similar in appearance. It is vital to have the best CNN models available that can be linked into a machine-vision system to execute automated identification jobs in order to address the traceability problem associated with comparable machining components. The present study, as a result, examines the performance of several CNN models that may be incorporated into a machine-vision system to execute automated recognition tasks.

1.2 Problem Statement

The misclassification of components in a small- to medium-sized manufacturing organisation may have major repercussions for the company's bottom line. There are compromises in product traceability as a result of the way manual inspection is presently carried out in the industry. This situation makes it impossible to check the component number digitally as a consequence of the problem. A lack of current traceability ensures that customers continue to get faulty components, and the same events continue to take place. It is vital that manual inspections be converted as soon as feasible to digital and automated inspections as soon as practicable.

In recent years, artificial intelligence-based technologies have been employed to allow a clever and intelligent identification system for industrial machining components, which is becoming more popular. The efficiency of Convolutional Neural Networks (CNNs) has led to their widespread usage in image identification tasks, and they have gained in popularity as deep learning algorithms as a consequence of their use.

1.3 Research Question

Based on the problem statement, the following research questions can be listed as follow:

• How to perform recognition task of similar industrial machining parts using deep learning method?

• What is the performance analysis from the model trainig?

• What is the industrial applicability of transfer learning CNN models in performing the recognition task?

1.4 Research Objective

Based on the research questions, the following research objectives can be formulated as follow:

- To develop a framework using deep transfer learning method for recognition task.
- To analyze the performance of different transfer learning CNN models during performing the recognition task.

- To recognize the most efficient transfer learning CNN models during performing the recognition task.

1.5 Research Scope and Limitation

The project scopes are to analyze the performance of different transfer learning CNN models during performing the recognition task. The scope of the project was defined to collect and analyze the data for practical cases by using deep transfer learning method for recognition task. Scope and limitation of the current research works can be listed as follow:

1. The performance analysis of convolutional neural networks for recognition task of similar industrial machining parts is developed based on the Anaconda software and using the phython languages.
2. The data are analyzed by using the existing machine learning algorithm provided from real industrial issue.
3. The data from the developed platform and tools are limited to 2 types of model training that is MobileNet and NASNetMobile

1.6 Research Design and Planning

1.6.1 Research Framework

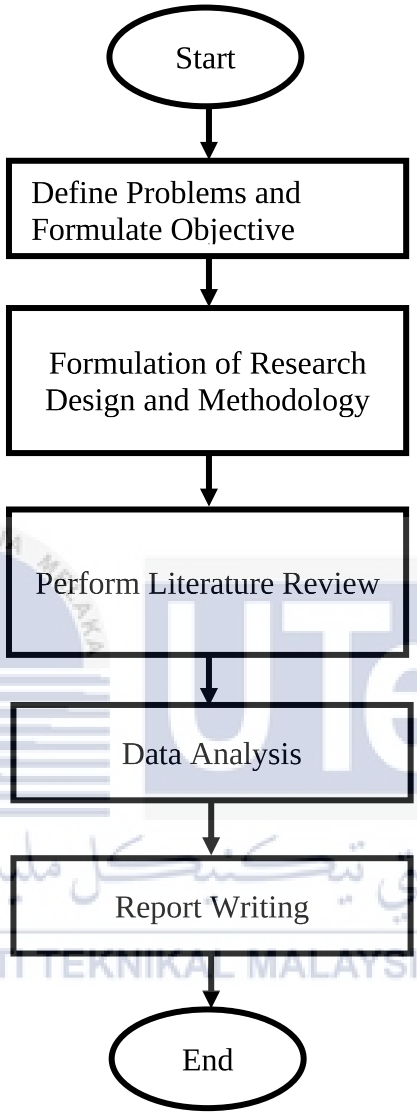


Figure 1 : Research Framework

Figure 1 shows the research frameworkin this semester. In this semester, the research work starts with defining the problem of the project and formulate the objectives. After that, formulation of research design, methodology, perform literature review, and collect preliminary data

1.6.2 Gantt Chart and Milestones

ACTIVITIES	STATUS	WEEK															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Meeting and discussion	Plan																
	Actual																
Conducting the experiment	Plan																
	Actual																
Collect data and make analysis on sample	Plan																
	Actual																
Discuss on result	Plan																
	Actual																
Start drafting and writing up report	Plan																
	Actual																
Start drafting and writing up chapter 4 and 5	Plan																
	Actual																
Recheck chapter 4 and 5	Plan																
	Actual																
Submission of first draft psm 2	Plan																
	Actual																
Submission of second draft psm 2	Plan																
	Actual																
Submission full report	Plan																
	Actual																
Finalize the correction of full report	Plan																
	Actual																
Preparation and presentation psm 2	Plan																
	Actual																

Table 1 Gantt chart of PSM 2

Table 1 present about the project activities throughout the whole semesters in form of a Gantt chart. The project starts on October 2021 until January 2022. The research work starts with defining the problem, formulation of research question and objective, designing the methodology, and lastly preliminary data collection. The research works end with a report writing for all the analysis and finding in the current project.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter provides a review of the literature on performance analysis of convolutional neural networks for recognition task of similar industrial machining parts. To begin, this chapter will discuss deep transfer learning, including deep learning, transfer learning, and convulutional neural networks (CNN) which is CNN mechanism and transfer learning CNN models.

2.2 Deep Transfer Learning

Machine-learning technology is used in a number of applications in modern culture, from online searches to content filtering on social media to product suggestions on e-commerce websites, and it is becoming more popular in consumer items like as cameras and smartphones. It is possible to employ machine-learning algorithms to recognise objects in photos, transcribe voice into text, match news articles, posts, or goods with users' interests, and choose relevant search results from a large number of search results, among other things. It is becoming more common to see deep learning methods being employed in various applications, and these approaches are becoming increasingly popular in general.

2.2.1 Deep Learning

Using deep-learning approaches, it is feasible to build many layers of abstraction by combining simple but non-linear modules that successively change the representation at one level (starting with the raw input) into a representation at a higher, somewhat more abstract level. It is feasible to learn enormously complicated functions by putting together a large number of transformations in a linear fashion. When performing classification tasks, it is necessary to emphasise characteristics of the input that are significant for discriminating while suppressing unimportant variations. Increased layers of representation are used to accomplish this. In the case of an image, the array of pixel values is represented as a first layer of representation, and the learned features in this layer generally reflect the presence or absence of edges at specified orientations and places in the picture, as seen in the figure. Most of the time, the second layer recognises motifs via the observation of certain configurations of edges, regardless of whether or not the edge placements are somewhat different from one another. When motifs are combined to build bigger combinations that match to components of recognisable items, the third layer may be able to recognise objects as combinations of these parts being constructed by succeeding layers. That these layers of characteristics are learnt from data rather than constructed by human engineers is the most essential part of deep learning. This is the cornerstone of deep learning because it is built on a general-purpose learning technique, which is the foundation of deep learning.(Lecun et al., 2015)

2.2.2 Transfer Learning

In machine learning, it is a sort of approach in which a model that has been trained on one job is used on a second task that is linked to the original task. A transfer learning approach is an optimization strategy that allows for quick progress or higher performance when modelling a second task after the first. Transfer learning is a study field for deep learning since it is connected to difficulties like as multi-task learning and idea drift, in addition to being a problem in itself. But even with the tremendous resources necessary to train Deep Learning models, transfer learning is popular because of the massive and demanding datasets on which deep learning models are trained, as well as the enormous resources required to train Deep Learning models. In deep learning, transfer learning is only effective when the model characteristics gained from the first task are generic.

2.3 Convolutional Neural Networks (CNN)

It is a deep learning algorithm known as a convolutional neural network (CNN) that is widely used in the field of image recognition. In artificial intelligence technology, CNNs can be thought of as a special type of feed-forward neural network. As opposed to its predecessors, One of CNN's most notable advantages is that it can automatically recognise key characteristics without the need for human involvement, making it the most frequently used machine learning system. (Dhillon & Verma, 2020; Yao et al., 2019). Unlike a typical multi-layer neural network, CNNs have the same number of layers as a standard multi-layer neural network, with at least one convolutional layer followed by at least one fully connected network layer at the bottom. Specifically, the CNN architecture employed in this study is built of the following components, with three convolutional layers and two fully connected layers being used in this approach, respectively.

2.3.1 CNN Mechanism

In CNN mechanism, the model serves as the foundation for the proposed CNN architecture.(Lecun et al., 2015) As shown in Fig. 3, the model is comprised of three convolutional layers followed by fully connected layers, with the convolutional layers acting as the first two layers of the model. The image of the machining parts was taken at a resolution of 750 x 1000 px and edited in Photoshop. It was necessary to downsize the original and enhanced photos in order to feed them into the CNN model. The images were downsized to a resolution of 224x224 pixels. Following that, the photos were converted to greyscale and the computer assigned them the dimensions 224x224 1 by default. Later, the greyscale pictures were subjected to a block of convolution layers with a kernel size of 3 x 3 and a stride of 1 pixel were added to them, after which they were processed once again..

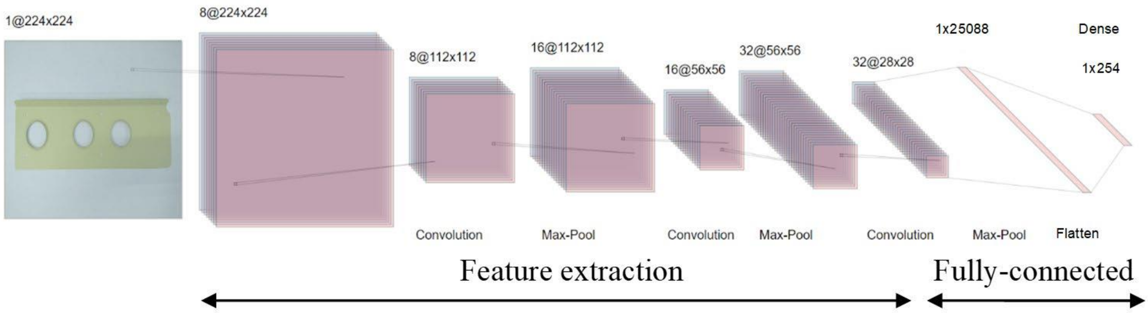


Figure 2 Architecture of the CNN model

Each of the three convolutional layers has an equal number of output filters, with the number of output filters in each of the three convolutional layers being 8, 16, and 32, respectively. Because of the convolutional layers, three max-pooling layers with window sizes of 2 by 2 and strides of 2 pixels each were added after them, with window sizes of 2 pixels and strides of 2 pixels each. This was done in order to reduce the spatial representation of the input data's spatial representation.(Liu & Deng, 2016) For the first time in the history of this approach, the Rectified Linear Unit (ReLU) function was employed as the activation function in the convolutional layer..

Conventional artificial neural networks are constructed around the fully connected layer, which serves as the network's core structural component. Using high-level filtering, it turns a picture of machined components into a number of votes that may be cast in real time. With this layer, the primary purpose is to do classification using the features collected by the previous convolutional layers as the input. Because the present work's class is binary, the model can only pick between two classes, Parts A and B. Because the current work's class is binary, the model can only choose between the two classes, Parts A and B. Flattening results in a single list being generated from the input data as a consequence of the procedure. The flattened layer has a height and width of 1 25088 pixels and a width of 0 pixels. In the next step, the flattened output is sent into a feed-forward neural network, where it is further processed.

To guarantee that the outputs of each training iteration in the dense layer are correct, the backpropagation method is utilised during each training iteration. The sigmoid activation function was used to the last layer of the CNN model in order to determine the likelihood that the sample belonged to each of the four classes in the dataset..

2.3.2 Transfer Learning CNN Models

When applied to machine learning, transfer learning is a challenging problem to solve. It retains the knowledge gained while solving one problem and applies it to a different but related problem later on in the same session. It is possible to apply the knowledge gained while learning to recognise cats when attempting to identify cheetahs, for example. Known as transfer learning, it is a deep learning technique that involves training a neural network model on a problem that is similar to the problem being solved before applying it to the actual problem at hand. As an added benefit, transfer learning can help to shorten the training time required to develop a learning model while also decreasing the generalisation error. For the model training we will go through the analysis of 6 models where for case 1 is MobileNet, and case 2 is NasNetMobile,

The model is trained by utilising both the enhanced and original datasets until the accuracy reaches a value of more than 95%.. Afterwards, a number of numerical tests are carried out to confirm the results. The model was run ten times, and the results were compared to one another. During the training and testing phases, the values for loss and accuracy per epoch were also determined. In order to evaluate the performance of the CNN model, it was necessary to calculate a confusion matrix for each numerical experiment performed. After that, the model was used to perform recognition and prediction tasks on a random picture from the test dataset, which was drawn from the training dataset. At long last, the model had been preserved, and the experiment had been successfully concluded. Training and recognition activities were performed a total of 10 times each. For each training step, the performance of the CNN model was assessed and shown in the form of a confusion matrix to demonstrate its effectiveness. The following equations may be used to compute the accuracy, precision, sensitivity, and specificity of the model based on this matrix::

$$\text{Accuracy}=(TN+TP)/(TN+FP+TP+FN)$$

$$\text{PPV}=TP/(TP+FP)$$

$$\text{NPV}= TN/(TN+FN)$$

$$\text{Sensitivity}=TP/(TP+FN)$$

$$\text{Specificity}=TN/(TN+FP)$$

2.4 Summary

This chapter examines the literature on the performance analysis of convolutional neural networks for the identification of comparable industrial machining parts, with a focus on the recognition of industrial machining parts. In deep-learning approaches, it is feasible to obtain several degrees of abstraction by building simple but non-linear modules that successively change the representation at one level into a higher, more abstract representation at a higher level of abstraction. A transfer learning approach is an optimization strategy that allows for quick progress or higher performance when modelling a second task after the first. As a result of the enormous and demanding data sets that deep learning models are trained on, transfer learning is becoming more prominent in deep learning.

The output of the flattened neural network is sent into a feed-forward neural network, where it is processed further. With this layer, the primary purpose is to do classification using the features collected by the previous convolutional layers as the input. In machine learning, transfer learning may be a difficult issue to tackle because of its large number of variables. Prior to applying the model to the real issue at hand, it is necessary to train it on a problem that is similar to that which is being addressed. Performance of the CNN model was tested and shown using a confusion matrix, which represented the model's performance.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter goes through the research methodology. The methodology is defined as a procedure or method that can be applied to meet the project's objectives in this study. This chapter focuses on designing and implementing a framework for recognising human behaviours using machine learning for factory worker monitoring. Data collection and information will be collected by using the android-based smartphone.

This chapter's content also explains how the present project's data collecting flow and methods work. The flow of data collection and process as well as steps for this project will be explained in detail. This chapter also includes an explanation of how to use the sensor on a smartphone to collect data from the accelerometer of human activities. The following research approach is chosen to ensure that the project follows the process flow.

3.1 Research Methodology

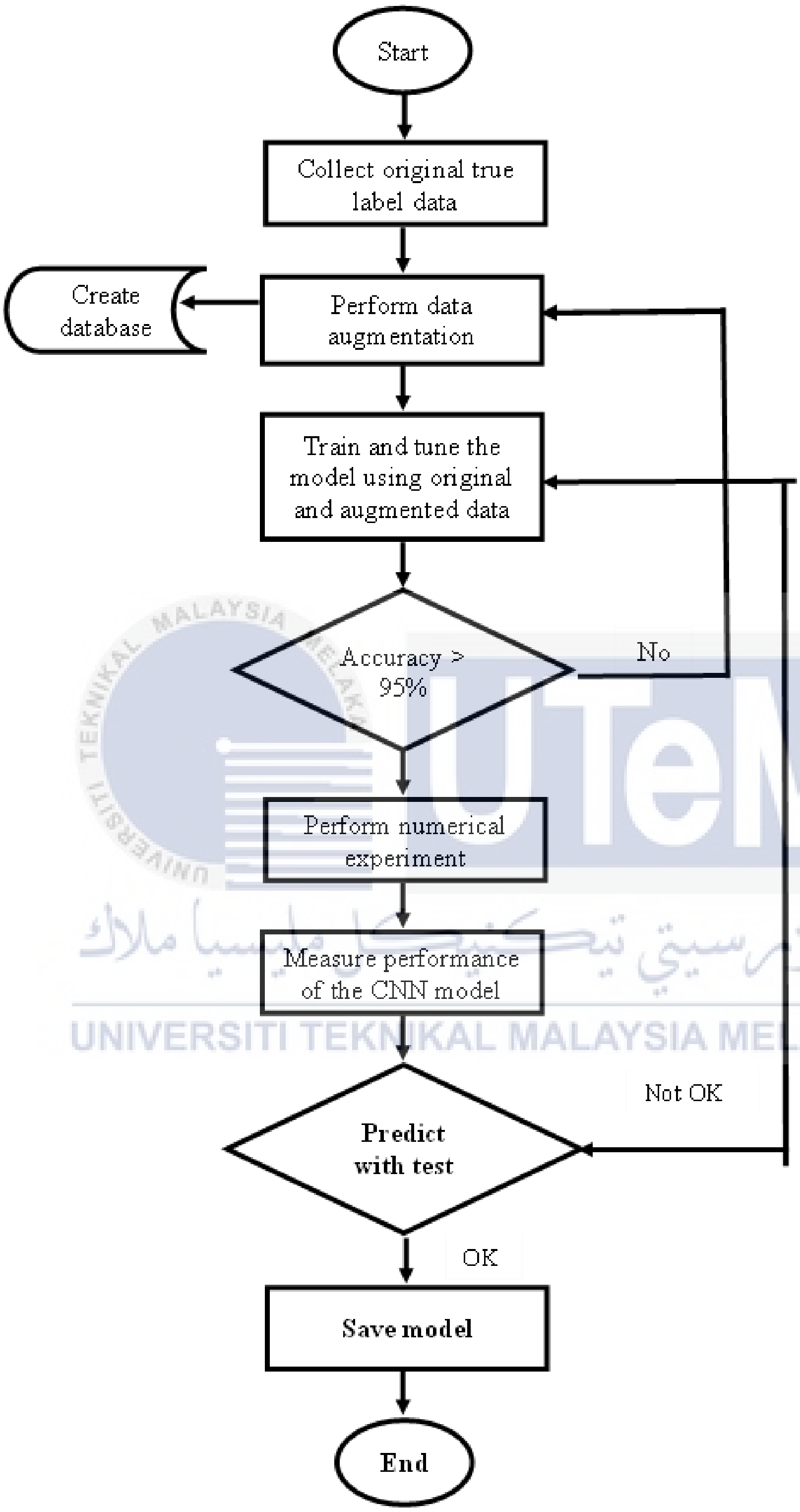


Figure 3 : Proposed research methodology

The research methodology in Figure 3.1, present about the method data collection from the dataset. The first step in the research methodology is to construct platform which refers to the training model. The selected training model to be used in this project is MobileNet and NasNetMobile and its task is to collect the performance data in form of accuracy. After that the machine learning algorithm will show performance analysis. From the analysis we can make the conclusion for the most efficient transfer learning CNN models during performing the recognition task.

3.1 Design of Experiment

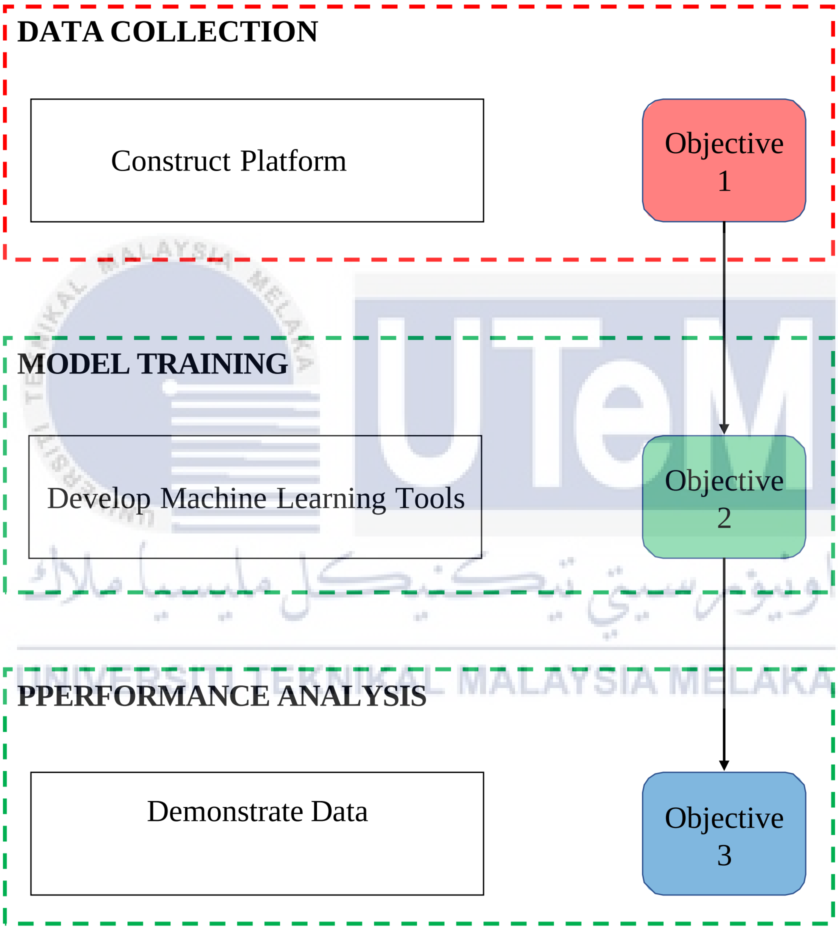


Figure 4 : Design of experiment

Identifying the objectives is an important step in a project to ensure that the project is aligned with the objective. Without a clear objective for the project, it will be difficult to make decisions what needs to be done in the study. Throughout this context, the title and the objective should be taken early in the run-up to the report. The research design and method as shown in Figure 3.2 present the project’s objectives and the associated research method to achieve it. The first objective is about data collection, followed by data analysis, and lastly preliminary data. This chapter will also explain in detail about each research design and method starting with the data collection.

3.1.1 Data Collection

The original photos of the machined components dataset, which had a resolution of 750 x 1000 pixels, were captured using an Android-based smartphone with a resolution of 1000 pixels. In order to complete both Parts A and B, a total of 160 pictures were taken, with each section including 80 shots.

Before being fed into the CNN model's input stream, the pictures are reduced in size to 224x224 pixels. Figure 1 shows an example of the resized original photographs obtained with an android-based smartphone for both Parts A and B, as well as the original images themselves. Figure 1: Original images taken with an android-based smartphone It can plainly be observed that the two regions are highly similar in terms of their natural environments. There is an exceedingly high likelihood that these two components will be misdiagnosed by a human operator in the future.

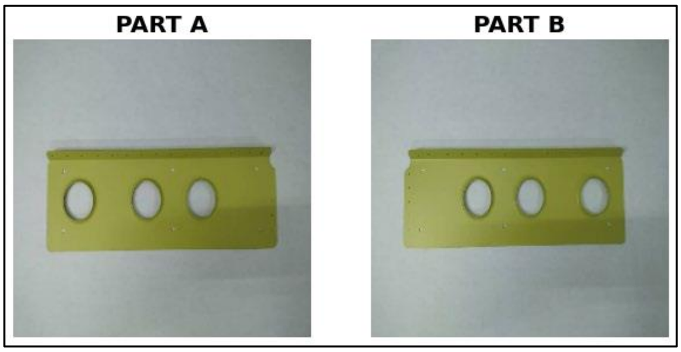


Figure 5 Sample of original images

According to the prior statement, the original photos were utilised to build an augmented dataset, which was then used to enhance the performance of the CNN model using the augmented dataset. For the final result, 2317 augmented photos were made by integrating several augmentation procedures such as rotation and translation with other processes such as magnification and brightness alteration to form a final output. The original and enhanced photos have been combined to generate a total of 2477 images, 1234 of which are in Part A and 1243 of which are in Part B. The original and augmented photographs have been combined to produce a total of 2477 images. It was necessary to blend the original and enhanced photos in order to create a total of 2477 photographs. The training dataset was constructed from 980 photographs from Part A, while the test dataset was constructed from the remaining 254 images from the same section. To produce the training dataset, a total of 1234 photos from Part A were utilised, while a total of 980 images from Part A were used to create the test dataset. Additionally, in Part B of the research, 987 photos were used to produce the training dataset, and 256 images were utilised to create the test dataset. For the purpose of training the CNN model, a balanced dataset was employed, with almost equal numbers of pictures being used for both training and testing of each class in both the training and testing stages, respectively. The data structure that was employed in the present investigation is shown in Figure 6.

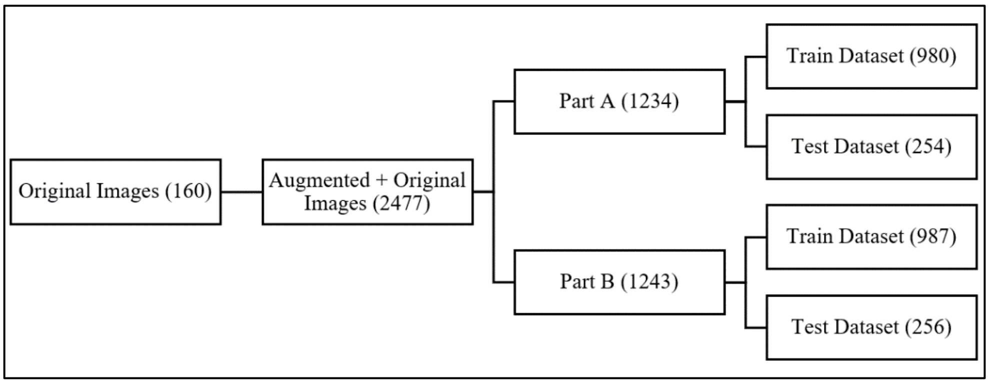


Figure 6 Data structure employed in the current work

3.1.2 Model Training

To achieve greater accuracy, all models are trained using both the augmented and original datasets until the accuracy reaches greater than 95%. Afterwards, a series of numerical experiments are carried out to confirm the results.The models were run a total of 30 times, and their performance was assessed. During the training and testing phases, the values for loss and accuracy per epoch were also determined. In order to evaluate the performance of the CNN model, it was necessary to compute a confusion matrix for each numerical experiment in addition. After that, the model was used to perform recognition and prediction tasks on a random image from the test dataset, which was drawn from the training dataset. At long last, the model had been saved, and the experiment had been successfully concluded.

3.1.3 Performance Analysis

Each training phase was followed by a measurement of the performance of the CNN model, which was then visualised in the form of a confusion matrix to illustrate its usefulness. Based on the information provided in this matrix, the following equations may be used to compute the accuracy, precision, sensitivity, and specificity of the model:

$$\text{Accuracy}=(TN+TP)/(TN+FP+TP+FN) \quad (1)$$
$$\text{PPV}=TP/(TP+FP) \quad (2)$$
$$\text{NPV}= TN/(TN+FN) \quad (3)$$
$$\text{Sensitivity}=TP/(TP+FN) \quad (4)$$
$$\text{Specificity}=TN/(TN+FP) \quad (5)$$

True positive, true negative, false positive, and false negative findings are denoted by the letters TP, TN, FP, and FN. When the CNN model correctly predicts Part A and the real label is identical to Part A, the value of TP in the confusion matrix shows the number of occasions in which the CNN model correctly predicts Part A and the true label is incorrect.

This statistic shows the number of occasions in which the CNN model has predicted that Part B is present, and the true label has been determined to be true in actuality. A positive FP value shows the number of instances in which a CNN model predicts a Part B, but the real label is actually Part A, while the preceding example's true label is Part A. In the end, the FN value indicates how many times the CNN model predicted that Part A was present while the real label was actually Part B.

The Precision Values for Parts A and B are sometimes referred to as the Positive Predictive Value (PPV) and the Negative Predictive Value (NPV), respectively, for the purposes of this definition. For the purpose of computing the PPV value, the number of observations that are anticipated to be positive (Part A) and are really positive is taken into consideration. As is true in other areas of mathematics, the Net Present Number (NPV) value reveals how many right forecasts there are out of all the wrong predictions (Part B). The Receiver Operating Characteristics (ROC) curve, as well as the Area Under the Curve (AUC) value, are also assessed throughout the training and testing operations. Additional information.

3.2 Summary

In order to ensure that the project is aligned with the goals, the identification of project objectives is an essential phase. It will be utilised for a variety of tasks, including data collecting and information gathering, on the Android-based smartphone. A detailed explanation of the data collecting and processing flow, as well as the procedures involved in this project, will be provided. A 224x224 pixel resolution is maintained throughout the picture resizing process before it is delivered into the CNN model's input stream. When several augmentation techniques, such as rotation, translation, zooming, and brightness modification, were combined, a total of 2317 enhanced pictures were generated in this research, indicating a successful outcome.

Combining the original and enhanced photos was essential in order to create a total of 2477 images from the original photographs. Next that, we looked at how well the CNN model performed throughout the training and testing stages of the experiment in the following section. It was then necessary to utilise a random picture from the test dataset, which was constructed by randomly picking photos from the training dataset, in order to evaluate the model's recognition and prediction skills on that image.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

In this chapter, we will focus on discussing the result performance analysis of convolutional neural networks for recognition task of similar industrial machining parts. A description of the result, out of several tests model trained performed and the calculation used will be discussed and explained through this chapter.

4.2 Data Collection

In this experiment, we will be using an android-based smartphone was used to capture the original images of the machining parts dataset, which had a resolution of 750 x 1000 pixels. It was necessary to take a total of 160 images for both Parts A and B, with each part containing 80 photographs.

Afterwards, the images are resized to 224x224 pixels before being fed into the CNN model's input stream. It can clearly be seen that these two sections are very similar in nature. An extremely high probability exists that these two components will be misidentified by a human operator. The original images were then used to generate an augmented dataset, which was then used to improve the performance of the CNN model, as previously stated

4.2.1 True Label Data Collection

First and foremost, true label data must be collected, which was accomplished by taking 160 images of Parts A and B with an Android-based smartphone. An augmented dataset was created by selecting images from among the original images at random. With this procedure, you will have an image database of machining parts, which can be used to train the CNN model later on. Following the collection of a large enough amount of data, the development of a CNN model is possible. FIGURE 3 shows a representation of the CNN model's architecture.

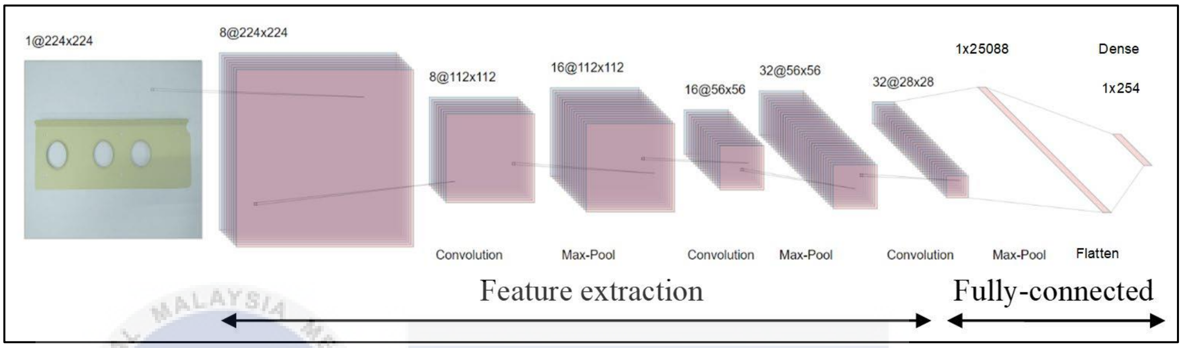


Figure 7Architecture of the CNN model

4.2.2 Data Augmentation

As an alternative to developing new photos and versions of existing ones, image data augmentation is a means of increasing a training dataset by including new images in the data set as well. This technique has the potential to increase the performance of deep learning models by producing variants of the pictures that they have been trained to recognise. Model overfitting is prevented by using data augmentation, which is a regulatory mechanism designed to keep models from getting overfit. The following procedures are carried out by this technique, as shown in Table 2.

Process	Description

Rotation	Randomly rotate image in the range of 00 to 450
Translation	Randomly shift image horizontally and vertically in the range of 0 to 0.1 (as a fraction of total width and height)
Zoom	Randomly zoom the image in the range of 0 to 0.2
Brightness	Randomly adjust image brightness in the range of 0.1 to 1.0

Table 2 Procedure works

A random selection of photos from the original dataset is used to construct the augmented dataset, which is then combined with the original dataset. The technique stated in Table 1 is then followed in order to obtain a total of 2317 enhanced pictures, which is the final result.

In order to build the enhanced dataset, a total of four processing steps were carried out on the original photos. In order to match the usual situation that occurs on a factory floor while conducting a recognition job on a production line, these operations have been selected. The pictures may be randomly put under the camera before to conducting the recognition job; as a consequence, the rotation and translation operations are employed to produce a series of enhanced images with random placement beneath the camera prior to doing the identification task. You may also modify the zoom and brightness of your camera in addition to these other features. It is thus necessary to use an enriched dataset that has a variety of zoom and brightness settings in order to train the model.

4.3 Model Training

For the model training we will go through of 2 models where for case 1 is MobileNet, and case 2 is NasNetMobile.

4.3.1 Case 1: MobileNet

To begin creating the model, I must first import all of the libraries that will be used to create the model into my project. It was necessary to define a data directory in the second step. The data that was discovered was revealed in the results. After that, the base model was created and then compiled together. The final result of the model compilation. Subsequently, the model was run 30 times with a train set of 30 epoch, and the data from the run was saved as an Excel spreadsheet in the format csv. We can obtain the results of the loss and accuracy comparisons, confusion metrics, ROC curves, and AUC values, as well as the result of the image cassification after the run is completed.


```
import os
import cv2
import warnings
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from matplotlib.image import imread

import tensorflow as tf
from keras import applications
from keras.applications import MobileNet
from keras import models, layers, optimizers
from keras.models import Model
from keras.layers import Input, Activation, Dropout, Flatten, Dense, Conv2D, MaxPooling2D
from keras.preprocessing import image
from keras.preprocessing.image import ImageDataGenerator, load_img, img_to_array, array_to_img
from keras.optimizers import Adam
from keras.callbacks import EarlyStopping, CSVLogger

from sklearn.metrics import classification_report,confusion_matrix,roc_curve,auc

%matplotlib inline
warnings.filterwarnings('ignore')
```

Figure 8 Imported libraries

```
#define datadirectory
my_data_dir = 'Desktop/PSM 2/assist_data/'
train_path = my_data_dir + 'train/'
test_path = my_data_dir + 'test/'

#define batch size
batch_size = 32

#create a data generator and rescale dataset
image_gen = ImageDataGenerator(rescale=1./255)

#load and iterate training dataset
train_set = image_gen.flow_from_directory(train_path,
                                          target_size=(224, 224),
                                          batch_size=batch_size,
                                          class_mode='binary',
                                          shuffle=True)

test_set = image_gen.flow_from_directory(test_path,
                                         target_size=(224, 224),
                                         batch_size=batch_size,
                                         class_mode='binary',
                                         shuffle=False)

#confirm the iterator works
batchX, batchy = train_set.next()
print('Batch shape=%s, min=%.3f, max=%.3f' % (batchX.shape, batchX.min(), batchX.max()))
```

Figure 9 Defined data directory

```
#check class label
train_set.class_indices
```

```
Found 2000 images belonging to 2 classes.
Found 500 images belonging to 2 classes.
Batch shape=(32, 224, 224, 3), min=0.000, max=1.000

{'D57443556232B': 0, 'D57443556233B': 1}
```

Figure 10 Class label

```
#create baseModel from keras applications
baseModel = MobileNet(weights="imagenet", include_top=False,
                  input_tensor=Input(shape=(224, 224, 3)))

#baseModel.summary()

#create headModel on top of baseModel
headModel = baseModel.output
headModel = Flatten(name="flatten")(headModel)
headModel = Dense(1024, activation='relu')(headModel)
headModel = Dropout(0.5)(headModel)
headModel = Dense(1, activation='sigmoid')(headModel)

#create a pretrained model
model = Model(inputs=baseModel.input,
              outputs=headModel,
              name='trained_mobilenet11')

#freeze the baseModel
for layer in baseModel.layers:
    layer.trainable = False

#set optimizer and compile the model
opt = Adam(lr=0.001)
model.compile(loss="binary_crossentropy",
              optimizer=opt,
              metrics=["accuracy"])
```

Figure 11 Base model

```
csv_logger = CSVLogger('log_11_MOBILENET.csv', append=True, separator=';')

results = model.fit(train_set,
                    validation_data=test_set,
                    epochs=30,
                    shuffle=True,
                    callbacks=[csv_logger])
```

Figure 12 Data saving

4.3.2 NasNetMobile

To begin creating the model, I must first import all of the libraries that will be used to create the model into my project. For the NasNetMobile the coding will use the same method in MobileNet model but the name of the model changed to NasNetMobile. It was necessary to define a data directory in the second step. The data that was discovered was revealed in the results. After that, the base model was created and then compiled together. The final result of the model compilation. Subsequently, the model was run 30 times with a train set of 30 epoch, and the data from the run was saved as an Excel spreadsheet in the format csv. We can obtain the results of the loss and accuracy comparisons, confusion metrics, as well as the result of the image cassification after the run is completed.

```python
import tensorflow as tf
from keras import applications
from keras.applications import NASNetMobile
from keras import models, layers, optimizers
```
Figure 13 Model name changes in libraries

```python
#create a pretrained model
model = Model(inputs=baseModel.input,
              outputs=headModel,
              name='trained_11_NASNetMobile')
```
Figure 14 Base model name change

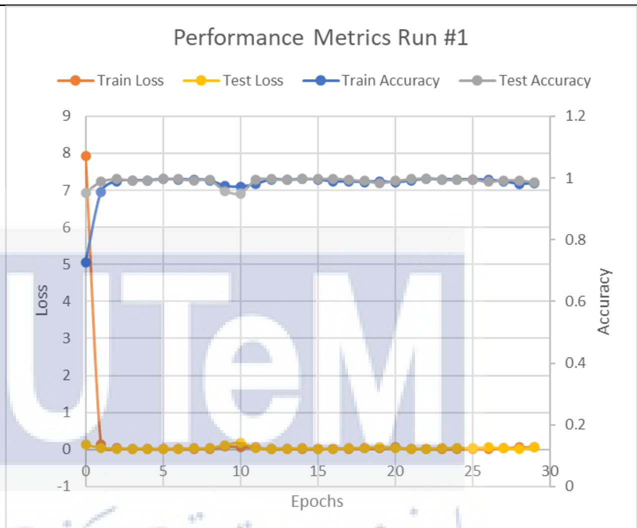
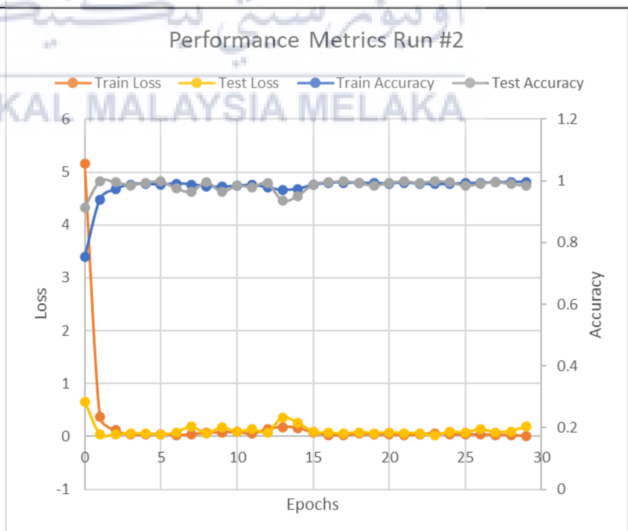
```python
csv_logger = CSVLogger('log_11_NASNetMobile.csv', append=True, separator=';')

results = model.fit(train_set,
                    validation_data=test_set,
                    epochs=30,
                    shuffle=True,
                    callbacks=[csv_logger])
```
Figure 15 Data model name change

4.4 Performance Analysis

The Confusion Matrix was used to assess the performance metrics of the CNN model under consideration. The Accuracy, Precision, Sensitivity, and Specificity values were calculated from the Confusion Matrix using the equations (1–5). Table 3, Table 4, Table 5, and Table 6 shows the results of the experiment from MobileNet and NasNetMobile.

4.4.1 Test Results

Within each of the following figures is a representation of how the loss and accuracy values changed during the course of the training operation. At various points in time throughout the experiment, the actual values of loss and accuracy were recorded for a total of 30 epochs. On the next figures, you can see the loss and accuracy diagrams, which represent the training process. Figures 1 and 2 show some preliminary results on the efficacy of the hyperparameters that have been chosen, as well as some further findings. Due to the fact that it is extensively utilised in binary classification problems and hence well-suited for this job, the binary cross-entropy loss function is used in the present study to solve the problem. In order to optimise the process, Adam (Adaptive Moment Estimation) was used, with a learning rate of 0.001 for the Adam algorithm that was used in the optimization process. Exact numbers of epochs and batch sizes of the CNN model were restricted for this experiment to thirty and thirty-two, respectively, in order to avoid overfitting.

Model	Performance
MobileNet Performance Metrices Run 1	
MobileNet Performance Metrices Run 2	

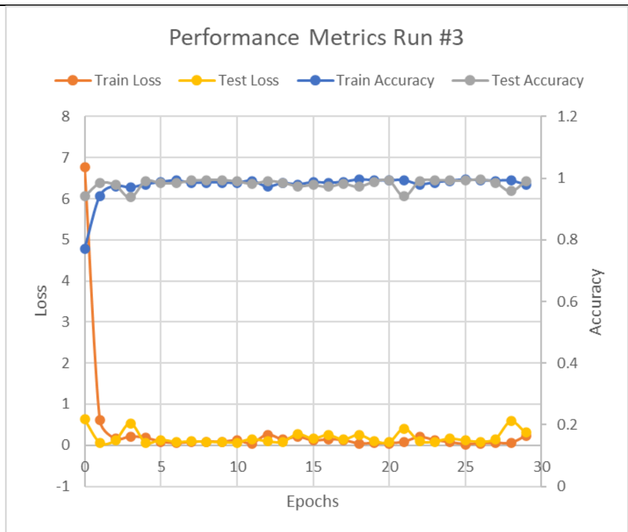
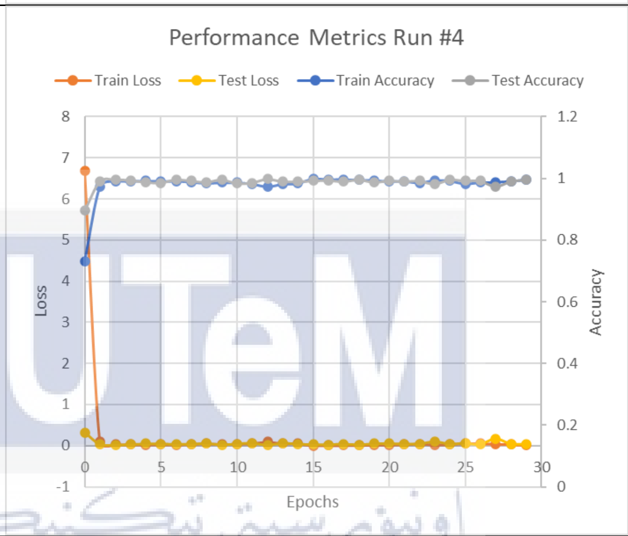
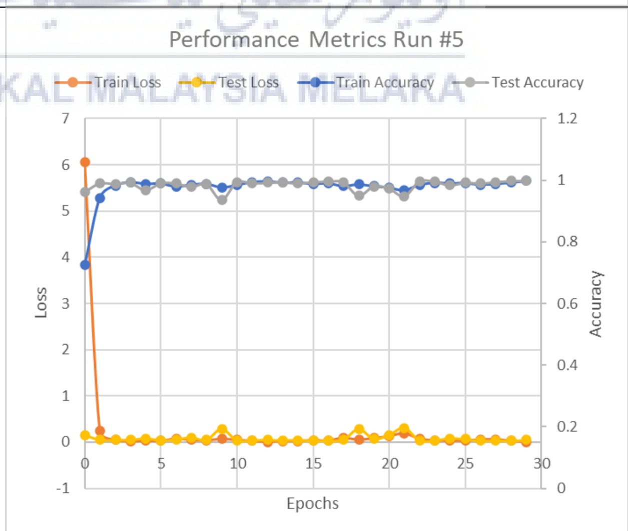
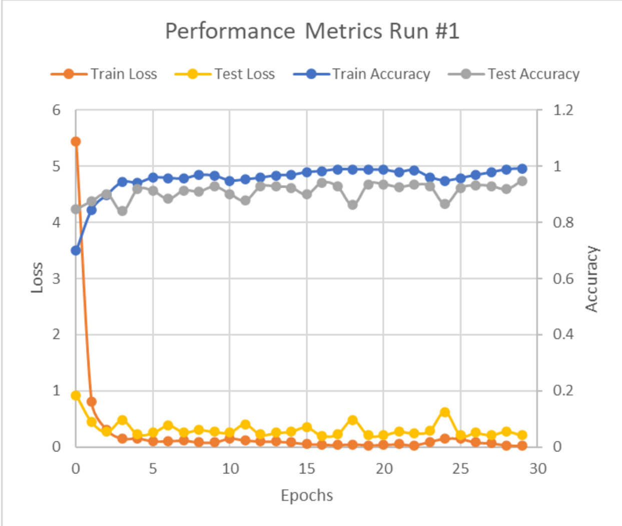
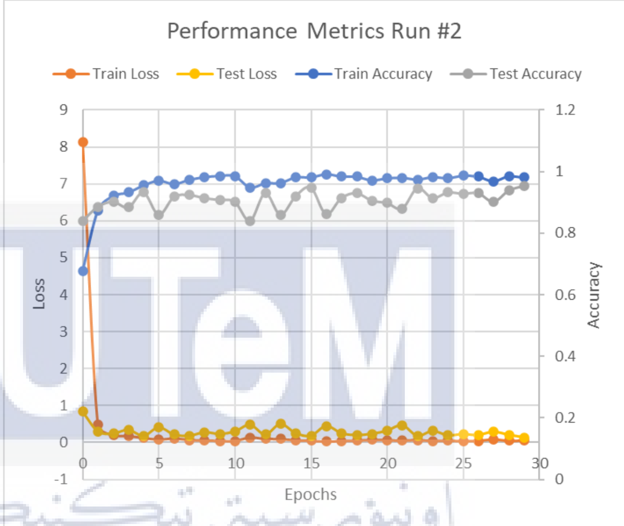
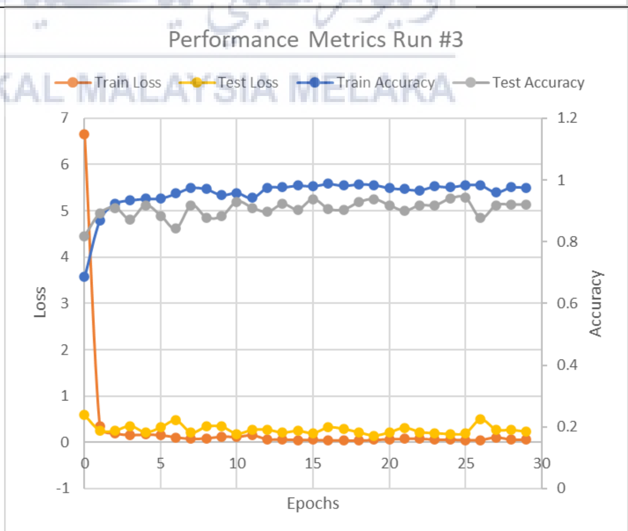
MobileNet Performance Metrices Run 3	
MobileNet Performance Metrices Run 4	
MobileNet Performance Metrices Run 5	

Table 3 MobileNet Performance Metrices

Model	Performance

<table>
<tr><td>NasNetNetMobile Performance Metrices 1</td><td>
</td></tr>
<tr><td>NasNetNetMobile Performance Metrices 2</td><td>
</td></tr>
<tr><td>NasNetNetMobile Performance Metrices 3</td><td>
</td></tr>
</table>

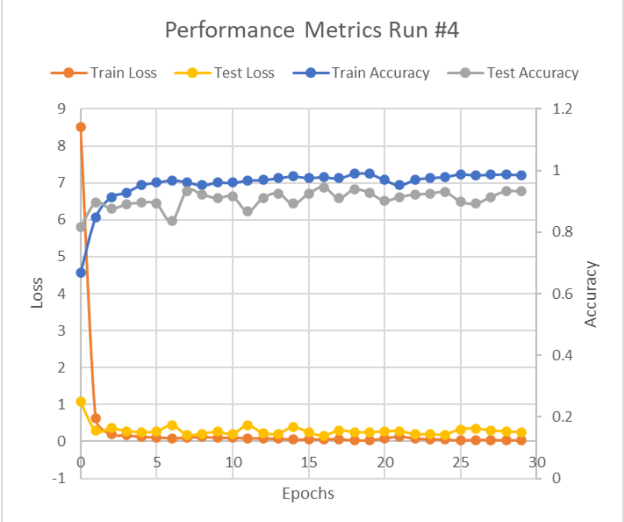
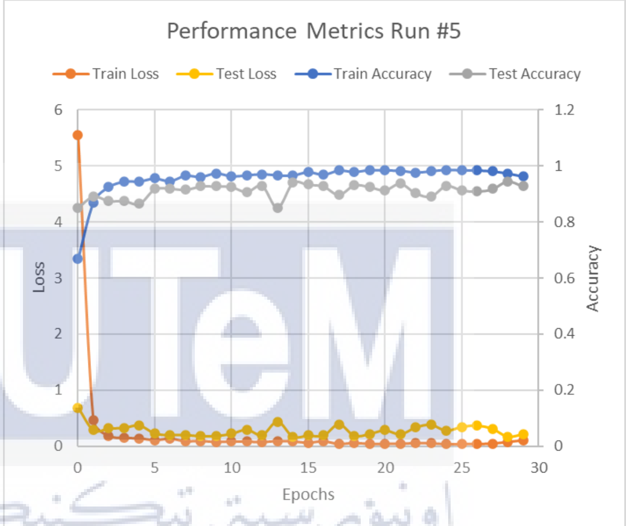
NasNetNetMobile Performance Metrices 4	
NasNetNetMobile Performance Metrices 5	

Table 4 NasNetNetMobile Performance

It can be seen in the graph that there is little difference between the training loss and test loss throughout the MobileNet experiment, whereas the NasNetMobile experiment revealed that there is a greater difference between the training loss and the test loss when compared to when compared to the MobileNet experiment. This shows the existence of a non-representative dataset, which suggests that the training dataset used to train the CNN model does not include enough information to address the recognition issue. This indicates the presence of a non-representative dataset. According to the information in Table 1, this event is compatible with the condition of the accuracy value that was achieved. The accuracy of Part A is lower than the precision of Part B, which can be noted when the accuracy of Part A and the precision of Part B are compared side by side. Thus, the training dataset for Part

A is just a marginally representative sample of the overall population, as shown above. The results of the other performance metric experiments will be included in the appendices.

4.4.2 Confusion Matrix

The Precision Values for Parts A and B are sometimes referred to as the Positive Predictive Value (PPV) and the Negative Predictive Value (NPV), respectively, for the purposes of this definition. For the purpose of computing the PPV value, the number of observations that are anticipated to be positive (Part A) and are really positive is taken into consideration. As is true in other areas of mathematics, the Net Present Number (NPV) value reveals how many right forecasts there are out of all the wrong predictions (Part B).

4.4.2.1 PERFORMANCE METRICES OF CNN MobileNet Model

Experiment	Confusion Matrix				Accuracy	Precision		Sensitivity	Specificity
	TP	TN	FP	FN		PPV	NPV		
1	288	245	3	4	0.987	0.990	0.984	0.986	0.988
2	250	242	1	7	0.984	0.996	0.972	0.973	0.996
3	247	248	4	1	0.990	0.984	0.996	0.996	0.984
4	249	249	2	0	0.996	0.992	1.000	1.000	0.992
5	250	249	1	0	0.998	0.996	1.000	1.000	0.996
6	249	248	2	1	0.994	0.992	0.996	0.996	0.992
7	250	238	1	11	0.976	0.996	0.956	0.958	0.996
8	250	237	1	12	0.974	0.996	0.952	0.954	0.996
9	250	248	1	1	0.996	0.996	0.996	0.996	0.996
10	249	249	2	0	0.996	0.992	1.000	1.000	0.992
11	248	247	3	2	0.990	0.988	0.992	0.992	0.988
12	248	248	3	1	0.992	0.988	0.996	0.996	0.988
13	250	245	1	4	0.990	0.996	0.984	0.984	0.996
14	250	243	1	6	0.986	0.996	0.976	0.977	0.996
15	250	248	1	1	0.996	0.996	0.996	0.996	0.996
16	250	247	1	2	0.994	0.996	0.992	0.992	0.996
17	250	236	1	13	0.972	0.996	0.948	0.951	0.996
18	248	247	3	2	0.990	0.988	0.992	0.992	0.988
19	250	249	1	0	0.998	0.996	1.000	1.000	0.996
20	248	247	3	2	0.990	0.988	0.992	0.992	0.988
21	251	241	0	0	1.000	1.000	1.000	1.000	1.000

22	250	244	1	5	0.988	0.996	0.980	0.980	0.996
23	249	249	2	0	0.996	0.992	1.000	1.000	0.992
24	250	248	1	1	0.996	0.996	0.996	0.996	0.996
25	249	248	2	1	0.994	0.992	0.996	0.996	0.992
26	249	249	2	0	0.996	0.992	1.000	1.000	0.992
27	250	237	1	12	0.974	0.996	0.952	0.954	0.996
28	250	246	1	3	0.992	0.996	0.988	0.988	0.996
29	250	248	1	1	0.996	0.996	0.996	0.996	0.996
30	250	246	1	3	0.992	0.996	0.988	0.988	0.996
Mean					0.990	0.994	0.987	0.988	0.994

Table 5 Performance Metrices of CNN MobileNet model

In accordance with the data obtained from all of the trials, the MobileNet CNN model has achieved the maximum accuracy value in the 21 experiments, with all of the values equal to one, according to the data. The results of this experiment showed that every component was accurately classified as either Part A or Part B. The MobileNet model has acquired the lowest accuracy value in the 17 experiments, which is the result of a combination of factors. A total of 13 Part B were wrongly recognised as Part A in this experiment, whereas just one Part A was incorrectly identified as Part B in the same experiment. Over the course of 30 tests, the MobileNet CNN model was able to attain an average accuracy of 0.990.

The number of photographs of Part A that were wrongly categorised as Part B is much larger than the number of images of Part B that were incorrectly classed as Part A. With another way of putting it, the precision value for Part A is greater than the precision value for the other part. This condition implies that the training dataset for Part B should be enhanced in some way.

4.4.2.2 PERFORMANCE METRICS OF CNN MobileNet Model

Experiment	TP	TN	FP	FN	Accuracy	Precision		Sensitivity	Specificity
						PPV	NPV		

1	243	234	8	15	0.954	0.968	0.940	0.942	0.967
2	231	235	20	14	0.932	0.920	0.944	0.943	0.922
3	249	217	2	32	0.932	0.992	0.871	0.886	0.991
4	249	217	2	32	0.932	0.992	0.871	0.886	0.991
5	233	231	18	18	0.928	0.928	0.928	0.928	0.928
6	250	196	1	53	0.892	0.996	0.787	0.825	0.995
7	247	200	4	49	0.894	0.984	0.803	0.834	0.980
8	242	225	9	24	0.934	0.964	0.904	0.910	0.962
9	216	243	35	6	0.918	0.861	0.976	0.973	0.874
10	239	222	12	27	0.922	0.952	0.892	0.898	0.949
11	248	215	3	34	0.926	0.988	0.863	0.879	0.986
12	244	222	7	27	0.932	0.972	0.892	0.900	0.969
13	248	210	3	39	0.916	0.988	0.843	0.864	0.986
14	245	228	6	21	0.946	0.976	0.916	0.921	0.974
15	241	235	10	14	0.952	0.960	0.944	0.945	0.959
16	243	217	8	32	0.920	0.968	0.871	0.884	0.964
17	241	226	10	23	0.934	0.960	0.908	0.913	0.958
18	247	218	4	31	0.930	0.984	0.876	0.888	0.982
19	229	239	22	10	0.936	0.912	0.960	0.958	0.916
20	243	220	8	29	0.926	0.968	0.884	0.893	0.965
21	248	213	3	36	0.922	0.988	0.855	0.873	0.986
22	243	233	8	16	0.952	0.968	0.936	0.938	0.967
23	247	212	4	37	0.918	0.984	0.851	0.870	0.981
24	250	202	1	47	0.904	0.996	0.811	0.842	0.995
25	249	198	2	51	0.894	0.992	0.795	0.830	0.990
26	249	208	2	41	0.914	0.992	0.835	0.859	0.990
27	248	215	3	34	0.926	0.988	0.863	0.879	0.986
28	241	218	10	31	0.918	0.960	0.876	0.886	0.956
29	241	224	10	25	0.930	0.960	0.900	0.906	0.957
30	248	200	3	49	0.896	0.988	0.803	0.835	0.985
Mean					0.924	0.968	0.880	0.893	0.967

Table 6 Performance metrics of CNN MobileNet model

According to the data obtained from all of the tests, the NasNet CNN model has achieved the greatest accuracy value in the first experiment, with a precision value for PPV of 0.968 and an NPV of 0.940 for the NPV value. The portions in this experiment were appropriately recognised as Part A and Part B, according to the results of the experiment. After six experiments, the NasNet model obtained the lowest accuracy value of 0.892, which is the lowest possible. In this experiment, 53 Parts B were wrongly recognised as Part A, whereas 1 Part A was incorrectly identified as Part B. The results of this experiment were as follows: The NasNeMobilet CNN model was able to obtain a mean accuracy of 0.924 over 30 tests using the data collected.

The number of photographs of Part A that were wrongly categorised as Part B is much larger than the number of images of Part B that were incorrectly classed as Part A. With another way of putting it, the precision value for Part A is greater than the precision value for the other part. This condition implies that the training dataset for Part B should be enhanced in some way.

4.4.2.3 Train and test accuracy for MobileNet

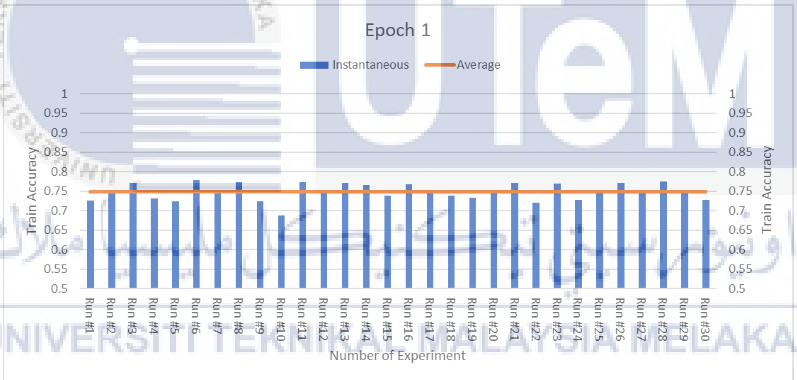


Figure 16 Epoch 1 Train accuracy for MobileNet

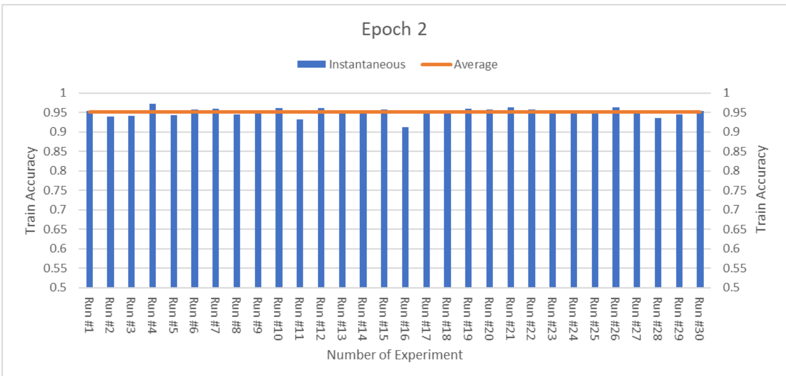


Figure 17 Epoch 2 Train accuracy for MobileNet

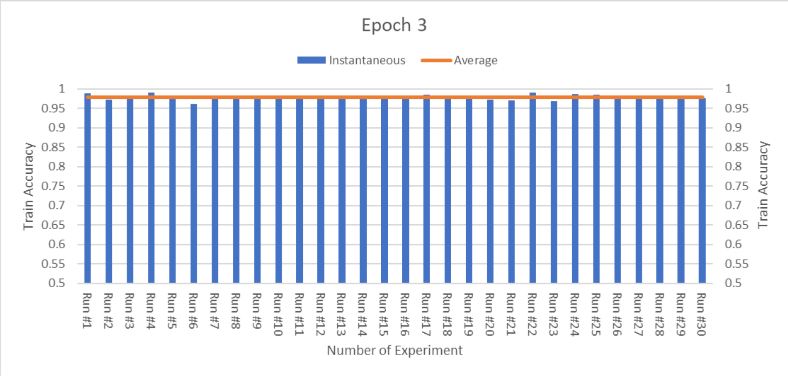


Figure 18 Epoch 3 Train accuracy for MobileNet

From the train accuracy for MobileNet Model, the graphs showed for 3 epochs. Its also showed that for epoch one, 19 run were has higher accuracy than average test accuracy. For epoch 2, it showed 25 run were has higher accuracy than average test accuracy and for epoch 3, it also showed 29 run has higher accuracy than average test accuracy. Its showed that the test accuracy getting better in every epoch. The other epoch will be showed in the appendices.

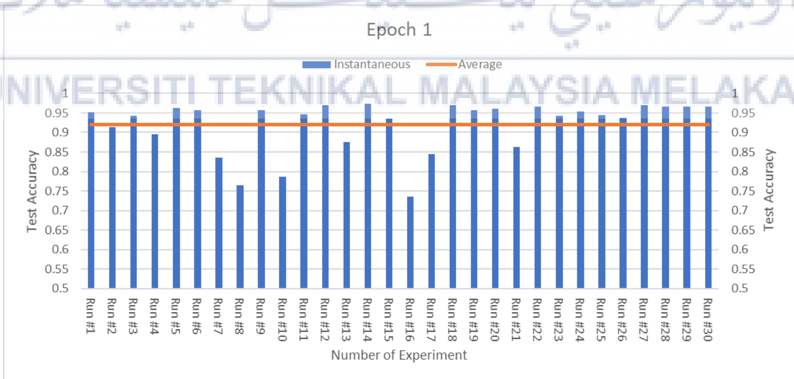


Figure 19 Epoch 1 Test accuracy for MobileNet Model

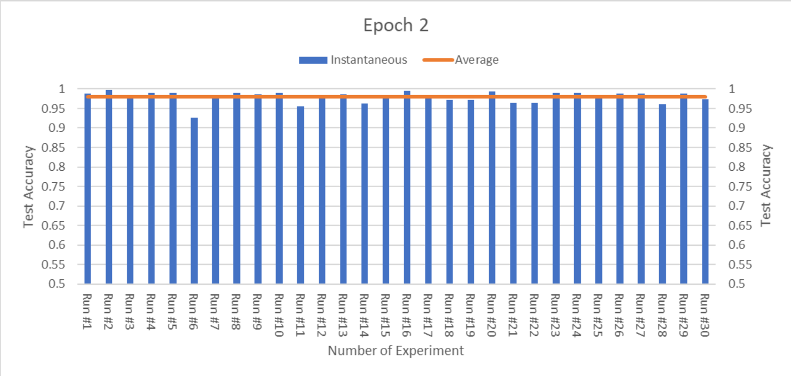


Figure 20 Epoch 2 Test accuracy for MobileNet Model

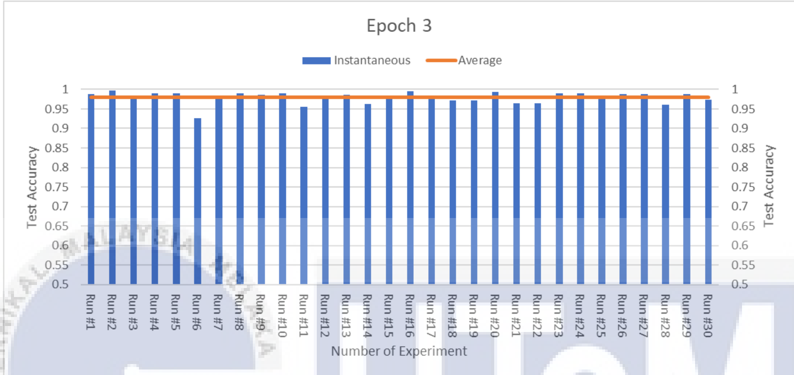


Figure 21 Epoch 3 Test accuracy for MobileNet Model

From the test accuracy for MobileNet Model, the graphs showed for 3 epochs. Its also showed that for epoch one, 21 run were has higher accuracy than average test accuracy. For epoch 2, it showed 23 run were has higher accuracy than average test accuracy and for epoch 3, it also showed 23 run has higher accuracy than average test accuracy. Its showed that the test accuracy getting better in every epoch. The other epoch will be showed in the appendices.

4.4.2.4 Train and Test accuracy for NasNet

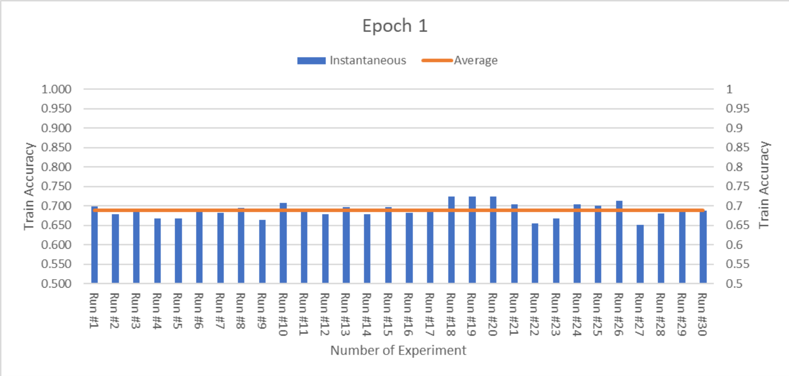


Figure 22 Epoch 1 Train accuracy for NasNet

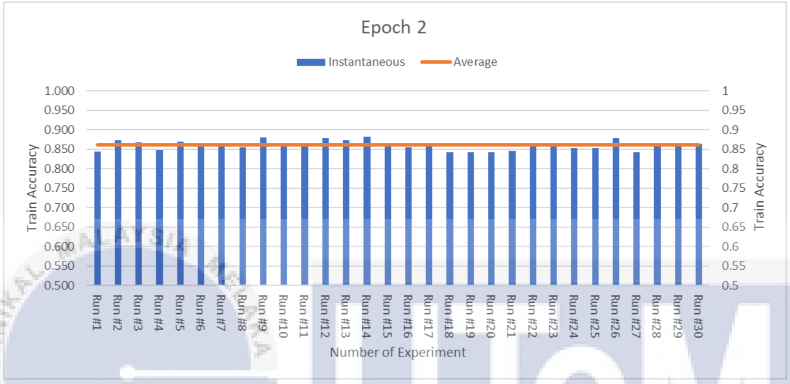


Figure 23 Epoch 2 Train accuracy for NasNet

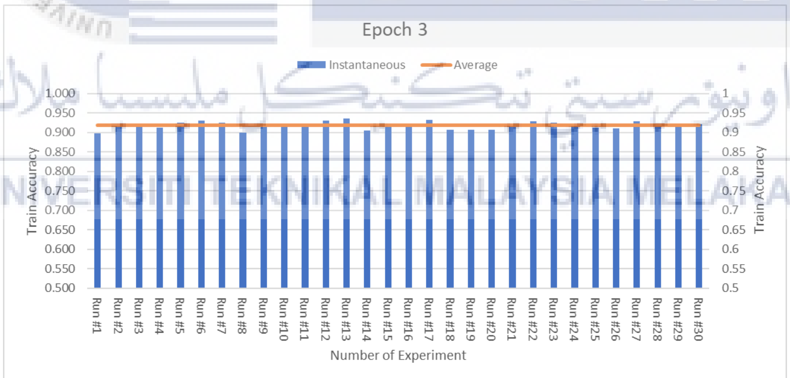


Figure 24 Epoch 3 Train accuracy for NasNet

From the test accuracy for MobileNet Model, the graphs showed for 3 epochs. Its also showed that for epoch one, 20 run were has higher accuracy than average test accuracy. For epoch 2, it showed 23 run were has higher accuracy than average test accuracy and for epoch 3, it also showed 24 run has higher accuracy than average test accuracy. Its showed that the test accuracy getting better in every epoch. The other epoch will be showed in the appendices.

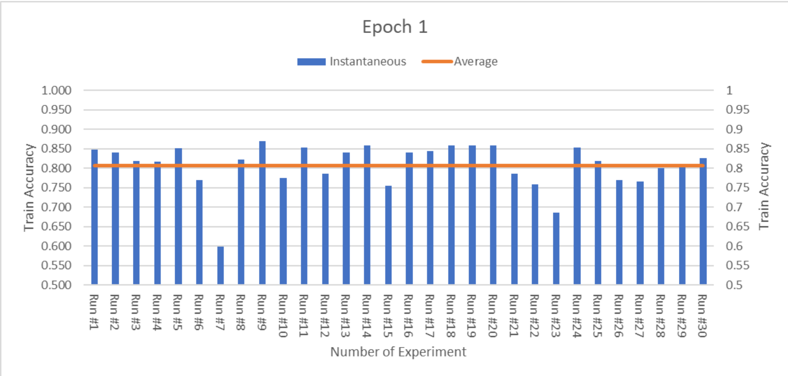


Figure 25 Epoch 1 Test accuracy for NasNet

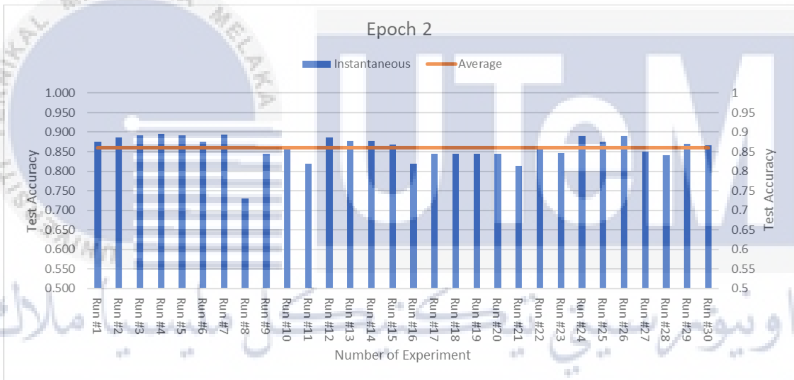


Figure 26 Epoch 2 Test accuracy for NasNet

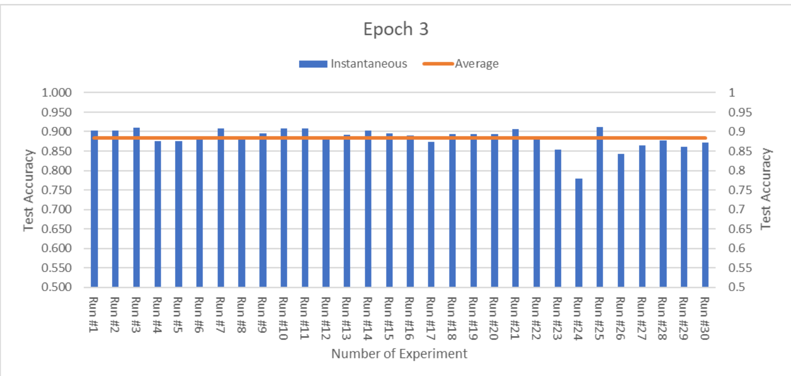


Figure 27 Epoch 3 Test accuracy for NasNet

From the test accuracy for MobileNet Model, the graphs showed for 3 epochs. Its also showed that for epoch one, 20 run were has higher accuracy than average test accuracy. For epoch 2, it showed 18 run were has higher accuracy than average test accuracy and for epoch 3, it also showed 32 run has higher accuracy than average test accuracy. Its showed that the test accuracy getting better in every epoch. The other epoch will be showed in the appendices.

4.5 Summary

In this chapter, we will look at the performance analysis of convolutional neural networks for the job of recognising comparable industrial machining components that are similar to one another. This chapter will include a summary of the results, which were obtained through a series of tests that the model trained had completed, as well as an explanation of the calculations that were employed. The training of the model necessitates the use of an enhanced dataset with a variety of zoom and brightness settings. A random selection of photos from the original dataset is used to construct the enhanced dataset. For the purpose of creating a sequence of augmented photos, it is possible to randomly put the images beneath the camera prior to executing the identification job.

The information that was uncovered was made public in the findings of the study. The findings of the experiment from MobileNet and NasNetMobile are shown in the following tables: Table 1, Table 2, and Table 3. On the next figures, you can see the loss and accuracy diagrams, which represent the training process. Part A's accuracy is lower than Part B's precision, and vice versa. From the accuracy data we can compare that MobileNet model is more efficient than NasNetMobile model since the data showed that the average mean accuracy of 99% compared to 92% of average mean accuracy from NasNetMobile.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In this report, a CNN model was used to perform binary classification tasks on two machining parts that were like one another. The model is composed of three convolutional layers and three maxpooling layers for feature extraction, which are followed by two fully connected convolutional layers. Layers of recognition and classification are connected together. Two Parts A and B of the classification system have been assigned for this purpose. The dataset that was used to train the model contains 160 observations. There are 2317 augmented images in addition to the original images. There are four different types of Processes of data augmentation were used in order to improve the data. The model's ability to perform well Rotation, translation, and zooming are all possible. The augmentation included changes to the contrast, brightness, and other aspects of the image. process. After that, the images were assigned to one of two categories. Part A consists of 1234 images for training and 254 images for testing, as well as Part B (1243 images for training and 256 for testing).

The models was run 30 times, and the performance metrics (in the form of loss and accuracy values) were measured for each experiment. The results of the experiments were combined to form the final report. It was also necessary to record the confusion matrix, as well as the model's precision, sensitivity, specificity, among other things. According to the results of the experiments, the CNN model achieved a mean accuracy of 99 percent for the MobileNet model and 92 percent mean accuracy from NasNet mobile model. The results of the tests also reveal that the mean precision values for Parts A and B for MobileNet model are 0.994 and 0.987 and for NasNetMobile model are 0.968 and0.880, respectively. According to the results, the MobileNet CNN model is more effective and accurate compared to the NasNetMobile model at distinguishing between and classifying two similar objects. Additionally, the results demonstrate the analysis of part recognition between using two CNN models. It provides a compelling alternative to the manual inspections that are currently practiced in small- to medium-sized manufacturing enterprises (SMEs).

5.2 Recommendations

In future improvement,accuracy of the CNN model's results could be enhanced as follows:

I. should be compared to those of other well-known CNN architectures in recognising two identical machined parts, the findings of this study should be compared to those of other well-known CNN architectures, such as EfficientNetB0, DenseNet121, ResNet50, and VGG16.

II. Moreover, the analysis should use a better laptop or computer to get more accurate and faster model processing.

5.3 Project Potential

The study finding could be applied on CNN model development where this could help researchers to develop a more accurate CNN model on image recognition. for various studies.

REFERENCES

Bacioiu, D., Melton, G., Papaelias, M., & Shaw, R. (2019a). Automated defect classification of Aluminium 5083 TIG welding using HDR camera and neural networks. *Journal of Manufacturing Processes*, *45*(June), 603–613. https://doi.org/10.1016/j.jmapro.2019.07.020

Bacioiu, D., Melton, G., Papaelias, M., & Shaw, R. (2019b). Automated defect classification of SS304 TIG welding process using visible spectrum camera and machine learning. *NDT and E International*, *107*(June). https://doi.org/10.1016/j.ndteint.2019.102139

Cui, W., Zhang, Y., Zhang, X., Li, L., & Liou, F. (2020). Metal additive manufacturing parts inspection using convolutional neural network. *Applied Sciences (Switzerland)*, *10*(2). https://doi.org/10.3390/app10020545

Deshpande, A. M., Minai, A. A., & Kumar, M. (2020). One-shot recognition of manufacturing defects in steel surfaces. *Procedia Manufacturing*, *48*, 1064–1071. https://doi.org/10.1016/j.promfg.2020.05.146

Dhillon, A., & Verma, G. K. (2020). Convolutional neural network: a review of models, methodologies and applications to object detection. *Progress in Artificial Intelligence*, *9*(2), 85–112. https://doi.org/10.1007/s13748-019-00203-0

He, X., Wang, T., Wu, K., & Liu, H. (2021). Automatic defects detection and classification of low carbon steel WAAM products using improved remanence/magneto-optical imaging and cost-sensitive convolutional neural network. *Measurement: Journal of the International Measurement Confederation*, *173*(August 2020), 108633.

https://doi.org/10.1016/j.measurement.2020.108633

Ji, X., Yan, Q., Huang, D., Wu, B., Xu, X., Zhang, A., Liao, G., Zhou, J., & Wu, M. (2021). Filtered selective search and evenly distributed convolutional neural networks for casting defects recognition. *Journal of Materials Processing Technology*, *292*(January). https://doi.org/10.1016/j.jmatprotec.2021.117064

Jiang, L., Wang, Y., Tang, Z., Miao, Y., & Chen, S. (2021). Casting defect detection in X-ray images using convolutional neural networks and attention-guided data augmentation. *Measurement: Journal of the International Measurement Confederation*, *170*(August 2020), 108736. https://doi.org/10.1016/j.measurement.2020.108736

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Liu, S., & Deng, W. (2016). Very deep convolutional neural network based image classification using small training sample size. *Proceedings - 3rd IAPR Asian Conference on Pattern Recognition, ACPR 2015*, 730–734. https://doi.org/10.1109/ACPR.2015.7486599

Lv, X., Duan, F., Jiang, J. J., Fu, X., & Gan, L. (2020). Deep metallic surface defect detection: The new benchmark and detection network. *Sensors (Switzerland)*, *20*(6). https://doi.org/10.3390/s20061562

Ma, G., Yu, L., Yuan, H., Xiao, W., & He, Y. (2021). A vision-based method for lap weld defects monitoring of galvanized steel sheets using convolutional neural network. *Journal of Manufacturing Processes*, *64*(January), 130–139.

https://doi.org/10.1016/j.jmapro.2020.12.067

Mery, D. (2020). Aluminum Casting Inspection Using Deep Learning: A Method Based on Convolutional Neural Networks. *Journal of Nondestructive Evaluation*, *39*(1). https://doi.org/10.1007/s10921-020-0655-9

Miao, R., Shan, Z., Zhou, Q., Wu, Y., Ge, L., Zhang, J., & Hu, H. (2021). Real-time defect identification of narrow overlap welds and application based on convolutional neural networks. *Journal of Manufacturing Systems*, *January*, 1–11. https://doi.org/10.1016/j.jmsy.2021.01.012

Nagata, F., Tokuno, K., Mitarai, K., Otsuka, A., Ikeda, T., Ochi, H., Watanabe, K., & Habib, M. K. (2019). Defect detection method using deep convolutional neural network, support vector machine and template matching techniques. *Artificial Life and Robotics*, *24*(4), 512–519. https://doi.org/10.1007/s10015-019-00545-x

Nguyen, T. P., Choi, S., Park, S. J., Park, S. H., & Yoon, J. (2021). Inspecting Method for Defective Casting Products with Convolutional Neural Network (CNN). *International Journal of Precision Engineering and Manufacturing - Green Technology*, *8*(2), 583–594. https://doi.org/10.1007/s40684-020-00197-4

Pan, H., Pang, Z., Wang, Y., Wang, Y., & Chen, L. (2020). A New Image Recognition and Classification Method Combining Transfer Learning Algorithm and MobileNet Model for Welding Defects. *IEEE Access*, *8*, 119951–119960. https://doi.org/10.1109/ACCESS.2020.3005450

Perera, R., Guzzetti, D., & Agrawal, V. (2021). Optimized and autonomous machine learning framework for characterizing pores, particles, grains and grain boundaries in microstructural images. *Computational Materials Science*, *196*(May), 110524. https://doi.org/10.1016/j.commatsci.2021.110524

Snow, Z., Diehl, B., Reutzel, E. W., & Nassar, A. (2021). Toward in-situ flaw detection in laser powder bed fusion additive manufacturing through layerwise imagery and machine learning. *Journal of Manufacturing Systems*, *59*(January), 12–26. https://doi.org/10.1016/j.jmsy.2021.01.008

Staar, B., Lütjen, M., & Freitag, M. (2019). Anomaly detection with convolutional neural networks for industrial surface inspection. *Procedia CIRP*, *79*, 484–489. https://doi.org/10.1016/j.procir.2019.02.123

Westphal, E., & Seitz, H. (2021). A machine learning method for defect detection and visualization in selective laser sintering based on convolutional neural networks. *Additive Manufacturing*, *41*(February), 101965. https://doi.org/10.1016/j.addma.2021.101965

Yang, Y., Yang, R., Pan, L., Ma, J., Zhu, Y., Diao, T., & Zhang, L. (2020). A lightweight deep learning algorithm for inspection of laser welding defects on safety vent of power battery. *Computers in Industry*, *123*, 103306. https://doi.org/10.1016/j.compind.2020.103306

Yao, G., Lei, T., & Zhong, J. (2019). A review of Convolutional-Neural-Network-based action recognition. *Pattern Recognition Letters*, *118*, 14–22. https://doi.org/10.1016/j.patrec.2018.05.018

Zhang, Z., Wen, G., & Chen, S. (2019). Weld image deep learning-based on-line defects detection using convolutional neural networks for Al alloy in robotic arc welding.

Journal of Manufacturing Processes, *45*(June), 208–216. https://doi.org/10.1016/j.jmapro.2019.06.023

Bacioiu, D., Melton, G., Papaelias, M., & Shaw, R. (2019a). Automated defect classification of Aluminium 5083 TIG welding using HDR camera and neural networks. *Journal of Manufacturing Processes*, *45*(June), 603–613. https://doi.org/10.1016/j.jmapro.2019.07.020

Bacioiu, D., Melton, G., Papaelias, M., & Shaw, R. (2019b). Automated defect classification of SS304 TIG welding process using visible spectrum camera and machine learning. *NDT and E International*, *107*(June). https://doi.org/10.1016/j.ndteint.2019.102139

Cui, W., Zhang, Y., Zhang, X., Li, L., & Liou, F. (2020). Metal additive manufacturing parts inspection using convolutional neural network. *Applied Sciences (Switzerland)*, *10*(2). https://doi.org/10.3390/app10020545

Deshpande, A. M., Minai, A. A., & Kumar, M. (2020). One-shot recognition of manufacturing defects in steel surfaces. *Procedia Manufacturing*, *48*, 1064–1071. https://doi.org/10.1016/j.promfg.2020.05.146

Dhillon, A., & Verma, G. K. (2020). Convolutional neural network: a review of models, methodologies and applications to object detection. *Progress in Artificial Intelligence*, *9*(2), 85–112. https://doi.org/10.1007/s13748-019-00203-0

He, X., Wang, T., Wu, K., & Liu, H. (2021). Automatic defects detection and classification of low carbon steel WAAM products using improved remanence/magneto-optical imaging and cost-sensitive convolutional neural network. *Measurement: Journal of* *the International Measurement Confederation*, *173*(August 2020), 108633. https://doi.org/10.1016/j.measurement.2020.108633

Ji, X., Yan, Q., Huang, D., Wu, B., Xu, X., Zhang, A., Liao, G., Zhou, J., & Wu, M. (2021). Filtered selective search and evenly distributed convolutional neural networks for casting defects recognition. *Journal of Materials Processing Technology*, *292*(January). https://doi.org/10.1016/j.jmatprotec.2021.117064

Jiang, L., Wang, Y., Tang, Z., Miao, Y., & Chen, S. (2021). Casting defect detection in X-ray images using convolutional neural networks and attention-guided data augmentation. *Measurement: Journal of the International Measurement Confederation*, *170*(August 2020), 108736. https://doi.org/10.1016/j.measurement.2020.108736

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Liu, S., & Deng, W. (2016). Very deep convolutional neural network based image classification using small training sample size. *Proceedings - 3rd IAPR Asian Conference on Pattern Recognition, ACPR 2015*, 730–734. https://doi.org/10.1109/ACPR.2015.7486599

Lv, X., Duan, F., Jiang, J. J., Fu, X., & Gan, L. (2020). Deep metallic surface defect detection: The new benchmark and detection network. *Sensors (Switzerland)*, *20*(6). https://doi.org/10.3390/s20061562

Ma, G., Yu, L., Yuan, H., Xiao, W., & He, Y. (2021). A vision-based method for lap weld defects monitoring of galvanized steel sheets using convolutional neural network.

Journal of Manufacturing Processes, *64*(January), 130–139. https://doi.org/10.1016/j.jmapro.2020.12.067

Mery, D. (2020). Aluminum Casting Inspection Using Deep Learning: A Method Based on Convolutional Neural Networks. *Journal of Nondestructive Evaluation*, *39*(1). https://doi.org/10.1007/s10921-020-0655-9

Miao, R., Shan, Z., Zhou, Q., Wu, Y., Ge, L., Zhang, J., & Hu, H. (2021). Real-time defect identification of narrow overlap welds and application based on convolutional neural networks. *Journal of Manufacturing Systems, January*, 1–11. https://doi.org/10.1016/j.jmsy.2021.01.012

Nagata, F., Tokuno, K., Mitarai, K., Otsuka, A., Ikeda, T., Ochi, H., Watanabe, K., & Habib, M. K. (2019). Defect detection method using deep convolutional neural network, support vector machine and template matching techniques. *Artificial Life and Robotics*, *24*(4), 512–519. https://doi.org/10.1007/s10015-019-00545-x

Nguyen, T. P., Choi, S., Park, S. J., Park, S. H., & Yoon, J. (2021). Inspecting Method for Defective Casting Products with Convolutional Neural Network (CNN). *International Journal of Precision Engineering and Manufacturing - Green Technology*, *8*(2), 583–594. https://doi.org/10.1007/s40684-020-00197-4

Pan, H., Pang, Z., Wang, Y., Wang, Y., & Chen, L. (2020). A New Image Recognition and Classification Method Combining Transfer Learning Algorithm and MobileNet Model for Welding Defects. *IEEE Access*, *8*, 119951–119960. https://doi.org/10.1109/ACCESS.2020.3005450

Perera, R., Guzzetti, D., & Agrawal, V. (2021). Optimized and autonomous machine learning framework for characterizing pores, particles, grains and grain boundaries in microstructural images. *Computational Materials Science*, *196*(May), 110524. https://doi.org/10.1016/j.commatsci.2021.110524

Snow, Z., Diehl, B., Reutzel, E. W., & Nassar, A. (2021). Toward in-situ flaw detection in laser powder bed fusion additive manufacturing through layerwise imagery and machine learning. *Journal of Manufacturing Systems*, *59*(January), 12–26. https://doi.org/10.1016/j.jmsy.2021.01.008

Staar, B., Lütjen, M., & Freitag, M. (2019). Anomaly detection with convolutional neural networks for industrial surface inspection. *Procedia CIRP*, *79*, 484–489. https://doi.org/10.1016/j.procir.2019.02.123

Westphal, E., & Seitz, H. (2021). A machine learning method for defect detection and visualization in selective laser sintering based on convolutional neural networks. *Additive Manufacturing*, *41*(February), 101965. https://doi.org/10.1016/j.addma.2021.101965

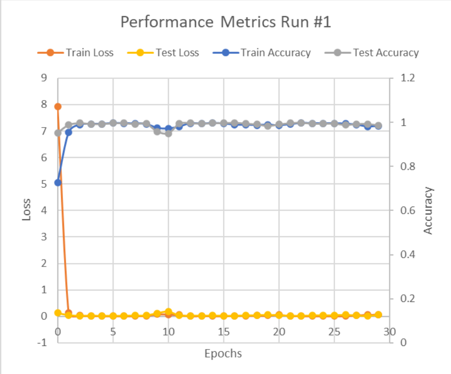
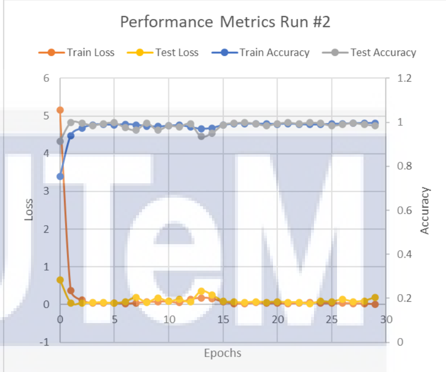
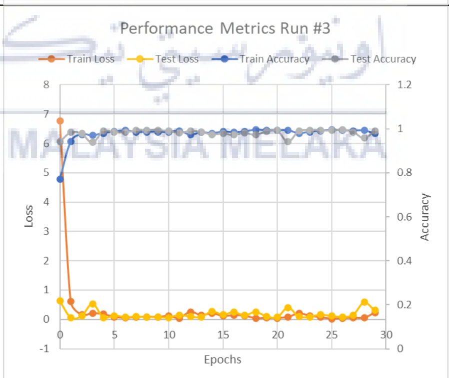
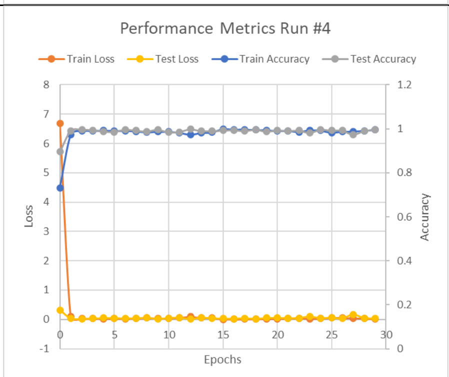
Yang, Y., Yang, R., Pan, L., Ma, J., Zhu, Y., Diao, T., & Zhang, L. (2020). A lightweight deep learning algorithm for inspection of laser welding defects on safety vent of power battery. *Computers in Industry*, *123*, 103306. https://doi.org/10.1016/j.compind.2020.103306

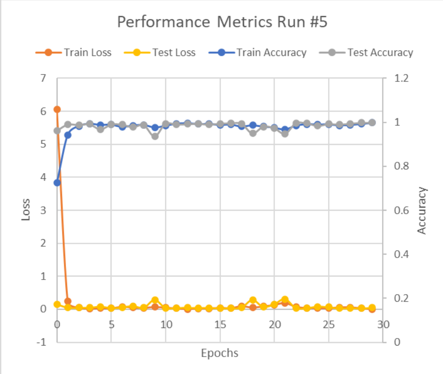
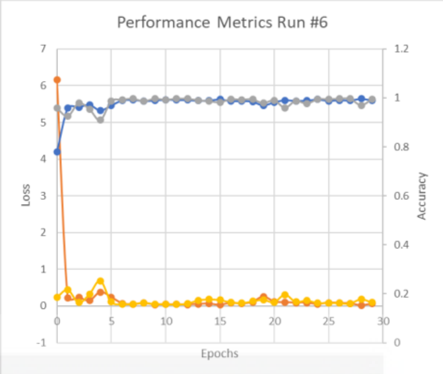
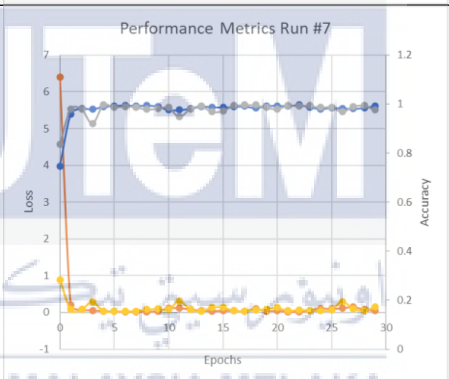
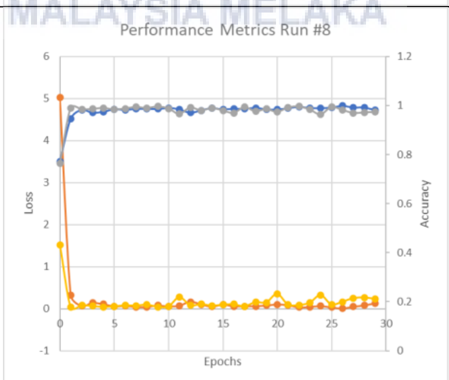
Yao, G., Lei, T., & Zhong, J. (2019). A review of Convolutional-Neural-Network-based action recognition. *Pattern Recognition Letters*, *118*, 14–22. https://doi.org/10.1016/j.patrec.2018.05.018

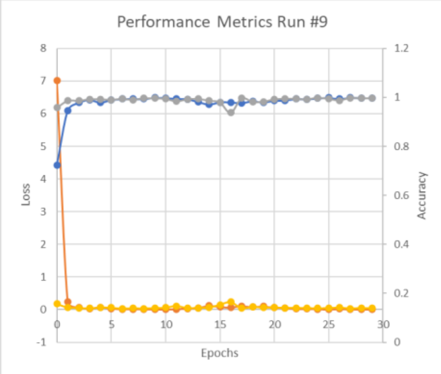
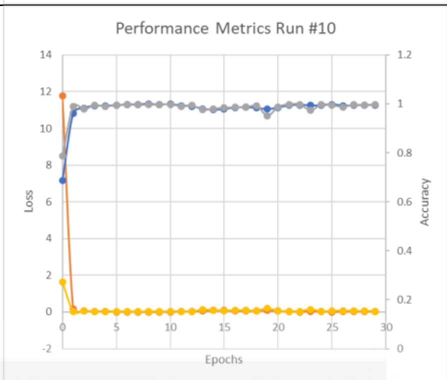
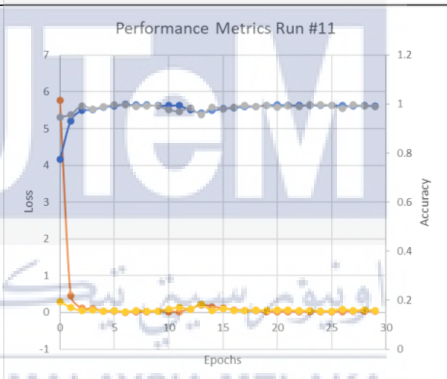
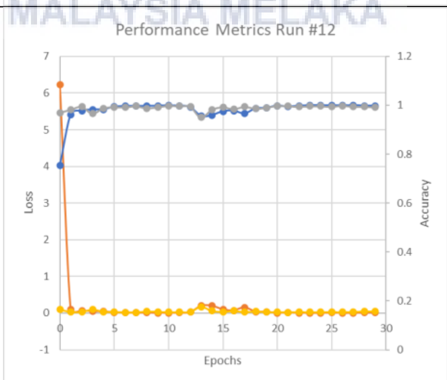
Zhang, Z., Wen, G., & Chen, S. (2019). Weld image deep learning-based on-line defects detection using convolutional neural networks for Al alloy in robotic arc welding. *Journal of Manufacturing Processes*, *45*(June), 208–216. https://doi.org/10.1016/j.jmapro.2019.06.023

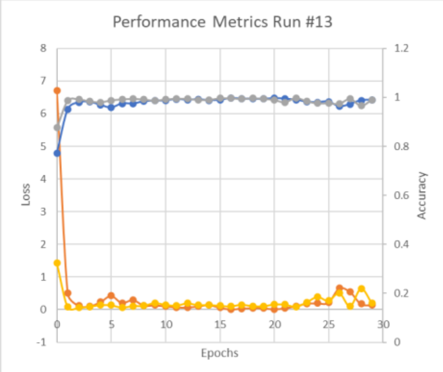
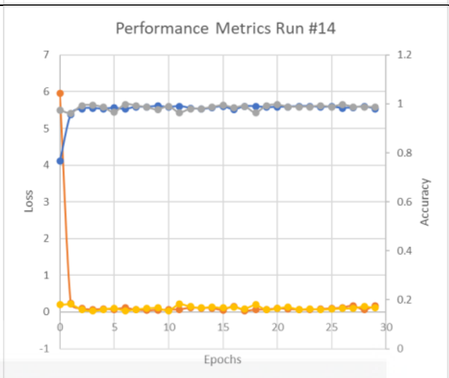
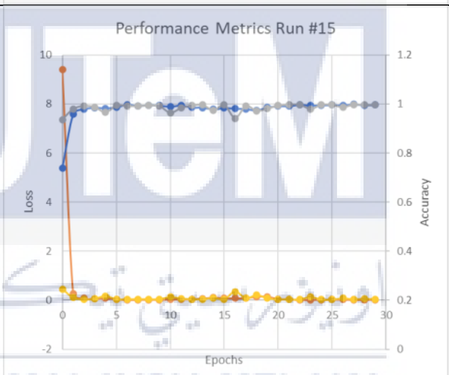
APPENDICES

APPENDIX A List of distribution Performance Metrices

Run	Result
1	
2	
3	
4	

5	 Performance Metrics Run #5 Train Loss Test Loss Train Accuracy Test Accuracy Loss Accuracy Epochs
6	 Performance Metrics Run #6 Loss Accuracy Epochs
7	 Performance Metrics Run #7 Loss Accuracy Epochs
8	 Performance Metrics Run #8 Loss Accuracy Epochs

9	 Performance Metrics Run #9 Loss Accuracy Epochs
10	 Performance Metrics Run #10 Loss Accuracy Epochs
11	 Performance Metrics Run #11 Loss Accuracy Epochs
12	 Performance Metrics Run #12 Loss Accuracy Epochs

<table>
<tr><td>13</td><td>
</td></tr>
<tr><td>14</td><td>
</td></tr>
<tr><td>15</td><td>
</td></tr>
</table>

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Universiti Teknikal Malaysia Melaka
Hang Tuah Jaya,
76100 Durian Tunggal,
Melaka, Malaysia.

+606 270 1000
+606 270 1022
www.utem.edu.my

FAKULTI TEKNOLOGI KEJURUTERAAN MEKANIKAL DAN PEMBUATAN
Tel : +606 270 1184 | Faks : +606 270 1064

Rujukan Kami (Our Ref):
Rujukan Tuan (Your Ref):
Tarikh (Date): 31 Januari 2021

Chief Information Officer
Perpustakaan Laman Hikmah
Universiti Teknikal Malaysia Melaka

Melalui

Dekan
Fakulti Teknologi Kejuruteraan Mekanikal dan Pembuatan
Universiti Teknikal Malaysia Melaka

Tuan

PENGKELASAN TESIS SEBAGAI TERHAD BAGI TESIS PROJEK SARJANA MUDA

Dengan segala hormatnya merujuk kepada perkara di atas.

2. Dengan ini, dimaklumkan permohonan pengkelasan tesis yang dilampirkan sebagai TERHAD untuk tempoh **LIMA** tahun dari tarikh surat ini. Butiran lanjut laporan PSM tersebut adalah seperti berikut:

Nama pelajar: MUHAMMAD FIKRI BIN FAHAR (B091310202)
Tajuk Tesis: PERFORMANCE ANALYSIS OF CONVOLUTIONAL NEURAL NETWORKS FOR RECOGNITION TASK OF SIMILAR INDUSTRIAL MACHINING PARTS

3. Hal ini adalah kerana IANYA MERUPAKAN PROJEK YANG DITAJA OLEH SYARIKAT LUAR DAN HASIL KAJIANNYA ADALAH SULIT.

Sekian, terima kasih.

"BERKHIDMAT UNTUK NEGARA"
"KOMPETENSI TERAS KEGEMILANGAN"

Saya yang menjalankan amanah,

DR HADYAN HAFIZH
Pensyarah Kanan
Jabatan Teknologi Kejuruteraan Pembuatan
Fakulti Teknologi Kejuruteraan Mekanikal dan Pembuatan
Universiti Teknikal Malaysia Melaka

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS LAPORAN PROJEK SARJANA

TAJUK: **PERFORMANCE ANALYSIS OF CONVOLUTIONAL NEURAL NETWORKS FOR RECOGNITION TASK OF SIMILAR INDUSTRIAL MACHINING PARTS**

SESI PENGAJIAN: **2021/22 Semester 1**

Saya **MUHAMMAD FIKRI BIN FAHAR**

mengaku membenarkan tesis ini disimpan di Perpustakaan Universiti Teknikal Malaysia Melaka (UTeM) dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hak milik Universiti Teknikal Malaysia Melaka dan penulis.
2. Perpustakaan Universiti Teknikal Malaysia Melaka dibenarkan membuat salinan untuk tujuan pengajian sahaja dengan izin penulis.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (✓)

☐	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia sebagaimana yang termaktub dalam AKTA RAHSIA RASMI 1972)
☐	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
√	TIDAK TERHAD	

Disahkan oleh:

Fikri

Alamat Tetap:

NO6 LOT 333 JALAN ESCORT,

KAMPUNG PINANG TAMBAHAN,

35900 TANJUNG MALIM PERAK.

Cop Rasmi:

Tarikh: 18 January 2022

Tarikh: 18 January 2022

** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh laporan PSM ini perlu dikelaskan sebagai SULIT atau TERHAD.