Addis Ababa University
Addis Ababa Institute of Technology
School of Mechanical & Industrial Engineering

Machine Learning Driven Quality Inspection for Cell Phone Assembly Process of a Manufacturing Company in Ethiopia

A Thesis Submitted to the School of Graduate Studies of Addis Ababa Institute of Technology, Addis Ababa University in partial fulfillment for the Degree of Master of Science in Industrial Engineering

By: Emebet Felege

Advisor: Ameha Mulugeta (Ph.D)

Addis Ababa, Ethiopia
June 2023

Addis Ababa University
Addis Ababa Institute of Technology
School of Graduate Studies
School of Mechanical and Industrial Engineering

“Machine Learning Driven Quality Inspection for Cell Phone Assembly Process of a Manufacturing Company in Ethiopia”

By

Emebet Felege

Approved by the Board of Examiners:

Dr. Ameha Mulugeta Advisor Name	Signature	Date
Dr. Ermias Tesfaye Internal Examiner	Signature	Date
Dr. Shimelis Tilahun External Examiner	Signature	Date
Dr. Araya Abera School Dean	Signature	Date
Dr. Sosina Mengistu Associate Director for PG Program	Signature	Date

Declaration

I hereby declare that the work which is being presented in this thesis entitled "Machine Learning Driven Quality Inspection for Cell Phone Assembly Process of a Manufacturing Company in Ethiopia" is original work of my own, has not been presented for a degree of any other university and all the resource of materials used for this thesis have been duly acknowledged.

Emebet Felege

Date

This is to certify that the above declaration made by the candidate is correct to the best of my Knowledge.

Dr. Ameha Mulugeta (Advisor)

Date

Acknowledgments

First and foremost, I would like to thank God for the gift of life and for giving me the strength and courage to complete this thesis. Next, I would like to thank my advisor, Dr. Ameha Mulugeta, for his guidance, support, and encouragement throughout this process. His insights and feedback have been invaluable to me. I would also like to thank Dr. Menore Tekeba, for his support related to machine learning issues and patience throughout the thesis work. In addition to his invaluable support, he has provided his GPU based laptop for my experiments. I am grateful for that.

I am also grateful to my family for their love and support. My parents have always been my biggest cheerleaders, and I could not have done this without them. My husband and daughters have also been a great source of support, and I am so lucky to have them in my life.

Finally, I would like to thank all of the people who have helped me along the way. My colleagues and friends have all played a role in this journey, and I am grateful for their contributions.

This thesis would not have been possible without the support of all of these people. I am truly grateful for their help.

Table of Contents

List of Figures

List of Tables

Abstract

In the process of cell phone component production, assembly and transportation, various types of defects will inevitably occur. Quality inspection in assembly process in time can avoid waste of manpower and material resources. In addition, it can monitor the production process, ensure high-quality products and reduce production costs. In this thesis work, the quality inspection problems in the cell phone assembly line of a manufacturing company in Ethiopia are studied. Most of the inspection techniques used in the company are manual based with inspectors check for defects using their eyes. The main challenges in the inspection process are detection accuracy and speed. In a one year inspection report, from the total defects found at the final inspection, more than 31 % are missed from receiving inspection. The manual visual inspection is also slow. For example, it takes at least three seconds to inspect a phone screen surface defects for a well-trained inspector. The inspection problems are categorized into five as image based, sound based, QR and barcode inspection, other functional tests and weight test. Based on the nature of the inspection problems and literature review, the image based and sound based problems are found to be best solved by machine learning (ML) driven solutions. Convolutional Neural Network (CNN) class ML models with their development procedure are proposed for image based problems while Recurrent Neural Network (RNN) and CNN class models are proposed for the sound based problems with the development procedures. To demonstrate this, five CNN class ML models are developed for phone screen inspection and their performances are compared. The dataset used for training and testing of these models is prepared by combining defect free images taken from the company with mobile phone screen surface defect segmentation (detection) dataset (MSD). The YOLOv8 model that is trained from scratch results in best performance. The validation accuracy is found to be 97.5 % and it takes 25 milliseconds to inspect surface defect of a phone screen. This result suggests the potential of ML driven approaches for solving the inspection problems of the case company. By taking defective and defect free images of phone screen from the company, a dataset is prepared to train a pre-trained YOLOv8 model. The validation accuracy is found to be 100%, but this model is prone to error threat. This is because the size of the dataset used is very small.

Keywords; Machine Learning, Quality Inspection, Cell Phone Assembly, Screen Surface Defect

Chapter 1 Introduction

1.1 Background

Quality control is essential for ensuring that products meet specific requirements and specifications. This is important for businesses because it helps to build trust with customers, boost customer loyalty, and reinforce brand reputation.

The saturation of established markets and the high competition in the development of new markets have led companies to optimize their production. Quality has become one of the key factors for manufacturers, so they must ensure the quality of their products in order to stand out from the competition (Reichenstein et al., 2022).

Quality control and inspection are essential for ensuring the long-term success of a manufacturing company. This is because a comprehensive and reliable quality inspection system can help to prevent low-quality products from reaching the market, which can save labor and resources. In addition, the results of the inspection can be used to identify and improve problematic process stages (C. Li et al., 2020).

Traditional quality inspection methods are often inefficient and do not meet the stringent requirements of today's competitive market. New tools and approaches are needed, such as machine learning-based solutions. Machine learning is a subfield of artificial intelligence that enables IT systems to recognize patterns and laws in data and develop solutions autonomously (Schmitt et al., 2020). This can help to improve the efficiency and performance of quality inspection processes.

Cell phones have become an indispensable part of people's lives, and the production quantity of global cell phones is increasing year by year. This makes it essential for cell phone manufacturers to improve quality and ensure production efficiency (Lei et al., 2018). Quality inspection in the production and assembly process can help to avoid waste of manpower and material resources, and it can also help to ensure high-quality products and reduce production costs.

In this thesis work, a cell phone assembly company in Ethiopia is taken. The company is engaged in the assembly of cell phones, televisions, battery chargers, and batteries. In order not to expose the company secret publicly, the name of the company is not mentioned in this document. In the cell phone assembly process of the company, there are a number of inspection points where quality can be checked. These include visual inspection, specific functional tests, physical requirements inspection, and packaging requirements inspection.

Visual inspection is used to check the appearance of the cell phone, including the components, the overall look, and the packaging. Functional tests are used to verify that the cell phone works as it should. Physical requirements inspection is used to check the weight of the product and packaging box. Packaging requirements inspection is used to check the graphics and logos on the packaging. The visual inspection covers the inspection of all the components before the assembling, during the assembling and the overall look of the cell phone after the assembly. The focus of the components visual inspection at the start of the assembly process is on the cell phone camera, screen and printed circuit board assembly (PCBA).

The functional tests are carried out to verify that the cell phone works the way it should, including powering device on and off, function of touch screen, display brightness and broken pixels, headphone output and ports.

For the physical requirements inspection, the main thing to look for is usually the weight of the product and packaging box. The weight of the product must be within an acceptable range of variation. The packaging requirements inspection is concerned with checking items such as the graphics and logos on the master carton, pallet packaging, and individual unit packaging.

The cell phone assembly company in Ethiopia that is the focus of this thesis uses traditional and manual-based quality inspection approaches. These approaches are inefficient and have poor performance, so state-of-the-art approaches are needed. Machine learning-based solutions are the best fit for most of the quality inspection problems in this company. However, in order to apply these solutions, a thorough study of the nature of the inspection problems should be performed.

1.2 Problem Statement

There are a number of quality inspection works at different stages of the cell phone assembly process in the company. From the site visit and assessment of the cell phone assembly line, the annual installed production capacity is 12,000,000. There are more than 25 components to be assembled for a single cell phone. All these components should be inspected for defects at different stages of the assembly process at the various inspection points. At each inspection point, the inspection problem (the parameters to be inspected) may not be single.

Most of the methods used for quality inspection are manual based. One of the manual ways to detect mobile defects is to arrange inspectors on the assembly line, where they use the naked eyes to detect the existence of defects on the phone components in order. However, with the growing demand for mobile phone, using the manual detection of component defects reveals many drawbacks. These include:

- Both detection accuracy and speed are low. There are many small defects on the phone screen, camera and PCBA that are difficult to detect through the human eyes, resulting in low accuracy of manual detection on the assembly line. In addition to that, the law accuracy is resulted from the large number of components that result fatigue and exhaustion to the inspectors.
- Lack of uniformity. Manual detection is mainly judged by human visual senses, which varies from different individuals. As a consequence, it is difficult to summarize a unified criterion. Different people may hold different test results on the same cell phone product.
- Labor cost is relatively high. Employing a large number of inspectors to detect defects on cell phone will greatly increase the company labor cost and reducing the market competitiveness. For example, in the first inspection point of an assembly line there are 12 inspectors. In addition to that, there is an additional cost of training for capacity building and new employees.

Such problems associated with the existing quality inspection approaches can be solved by employing machine learning based approaches.

1.3 Research Questions

The thesis work intends to answer the following research questions.

- What are the natures of the quality inspection problems in the cell phone assembly process?
- Which of the quality inspection problems are to be solved or improved by machine learning based approaches? And which are not?
- What kind of machine learning empowered quality inspection approaches can best support the company in eliminating drawback of the current inspection?

1.4 Objective

1.4.1 General Objective

The general objective of this thesis work is to study the existing quality inspection approaches used in the case company cell phone assembly process and to develop machine learning based quality inspection approaches that can improve the performance of the company.

1.4.2 Specific Objectives

The specific objectives include:

- To study the existing quality inspection approaches used in the overall assembly process and to understand the nature of the inspection problems.
- To identify which quality inspection problems are to be addressed by machine learning based approaches and which are not.
- To develop machine learning based approaches and solutions to the identified problems.
- To demonstrate the proposed solutions by developing a machine learning based model.

1.5 Scope

This thesis work is limited to the improvement of the quality inspection through machine learning driven approaches only for the cell phone assembly process of the case company. The model development is limited to only phone screen surface defect detection.

1.6 Limitation

Usually, problems solved by machine learning based approach are data intensive. In the case company, there is poor documentation and data handling. The limitation of this thesis work is the shortage of labeled data to be used for machine learning. This limitation is addressed by labelling data from scratch, by using similar dataset from other source and by using augmentation.

1.7 Significance

In the cell phone assembly process of the case company, there is a high volume of quality inspection. High inspection volumes turn inspection processes into manufacturing bottlenecks (Schmitt et al., 2020). The significance of the thesis work is to show how machine learning driven quality inspection approaches are developed and applied in relieving the cell phone assembly process from such problems.

Chapter 2 Literature Review

2.1 Quality Control and Quality Inspection

2.1.1 Quality Control

In today's competitive marketplace, it is essential for organizations to constantly change and adapt in order to maintain their market share. This is a challenge that all companies face, but some have adopted methods that emphasize quality engineering and customer satisfaction (Broday, 2022). Quality control is an essential and inevitable part of the process of manufacturing any product in today's industries. It encompasses all the activities involved in ensuring that products meet quality standards. This includes activities such as setting quality standards, monitoring the production process, and taking corrective action when necessary. Defect detection in any product is one of the most important quality control measures in the product manufacturing process (Sachan et al., 2020).

Quality control has undergone many changes and evolution in the past century (Zonnenshain & Kenett, 2020), (Broday, 2022). It has evolved from a focus on inspection to a more holistic approach that encompasses quality assurance, quality management, and quality by design (Zonnenshain & Kenett, 2020). These changes have been driven by advances in technology and the increasing complexity of products.

Some of the most famous quality initiatives adopted worldwide include Total Quality Management (TQM), Six Sigma, Lean Sigma, and Quality by Design. These quality movements were led by experts like Shewhart, Deming, Juran, Taguchi, and others who established a basis for the quality approach practiced in industry, business, and public service sectors (Zonnenshain & Kenett, 2020).

Although quality control has evolved significantly over the past century, there is still room for improvement. Some experts believe that the next generation of quality control models will need to incorporate artificial intelligence and other advanced technologies in order to meet the challenges of the future.

The fourth industrial revolution, also known as Industry 4.0, is characterized by the use of advanced technologies such as big data, artificial intelligence, and robotics. These technologies have the potential to revolutionize the way quality is managed in manufacturing.

One of the key aspects of Quality 4.0 is the use of real-time data. This data can be used to monitor the production process and identify any deviations from the established quality standards. This allows for problems to be identified and addressed quickly, which can help to prevent defects from occurring.

Another important aspect of Quality 4.0 is the use of artificial intelligence. AI can be used to analyze large amounts of data and identify patterns that would not be possible to detect with human eyes. This information can be used to improve the quality of products and processes (Broday, 2022).

The future of quality control is closely linked to the development of Industry 4.0. As these technologies become more widespread, they will provide new opportunities for improving the quality of products and processes.

However, there are also some challenges that need to be addressed. For example, the use of big data and artificial intelligence can be expensive. Additionally, these technologies can be complex to implement and manage.

The challenges facing quality control are not insurmountable, and the future of the field is bright. By embracing the latest technologies, organizations can improve the quality of their products and processes, which will lead to increased customer satisfaction and profitability.

2.1.2 Quality Inspection

Quality inspection is a critical process in manufacturing that ensures that products meet the defined quality standards. It is typically carried out by a team of quality inspectors who measure, examine, test, or gage the product's characteristics to see if they conform to the original design. If a product does not meet the standards, it is rejected and either repaired or scrapped.

This process is vital because it eliminates the risk of supplying faulty parts to customers and can provide valuable insights that help to address manufacturing process issues. It is an important component of today's smart manufacturing systems, which are aligned with the development of Industry 4.0 (Babic et al., 2021). This article reviews a wide variety of quality inspection systems that utilize different technologies, ranging from human operators to high-fidelity sensor systems to image-based systems.

There are three main types of inspection systems: manual, semi-automatic, and automated (Rahman et al., 2019).

- Manual inspection is fully dependent on human judgment. The inspector uses their eyes, hands, and tools to measure, examine, test, or gage the product's characteristics to see if they conform to the original design.
- Semi-automatic inspection systems consist of automated mechanisms that are introduced during the inspection by the operator to expedite the manual inspection. For example, the operator may use a laser pointer to identify defects, and the machine will then automatically measure the defect.
- Automated inspection systems run automatically with the presence of the operator, only to monitor the machine performance. These systems use a variety of sensors and cameras to measure and analyze the product's characteristics.

The type of inspection system that is used depends on the specific needs of the manufacturing process. For example, manual inspection may be sufficient for simple products, while automated inspection may be necessary for complex products.

There are a variety of automated inspection systems available, each with its own strengths and weaknesses. Some of the most common types of automated inspection systems include (Babic et al., 2021):

- Visual testing: This type of system uses cameras to capture images of the product and then analyzes the images to look for defects.
- Sensor-based systems: These systems use sensors to measure the product's characteristics, such as its dimensions, weight, or temperature.
- Magnetic particle inspection: This type of system uses magnetic fields to detect defects in ferromagnetic materials.
- Radiographic testing: This type of system uses X-rays to create images of the product's interior, which can be used to detect defects.
- Ultrasound testing: This type of system uses sound waves to create images of the product's interior, which can be used to detect defects.

- Penetrant testing: This type of system uses a dye to detect surface defects in non-ferromagnetic materials.

Among these methods, image-based systems are among the most popular because they can be implemented relatively easily and at a lower cost. They are also non-contact, which means that they do not damage the product.

Automated inspection systems can significantly improve product quality and production rates in manufacturing (Miles & Surgenor, 2011). Machine vision (MV) is one such technology that has been successfully applied to production lines. In recent years, MV technology has become increasingly popular in industrial production inspection due to its non-contact measurement, high accuracy, and efficiency (Xiao et al., 2022).

2.1.3 Acceptance Sampling and Acceptance Quality Level

Acceptance sampling is a statistical method used in quality control to determine the quality of an entire batch of products by inspecting a randomly selected sample. The Acceptable Quality Level (AQL) is the worst quality level that is still considered acceptable. It is typically expressed as a percentage of defective units in a batch.

Acceptance sampling is used in a wide variety of industries, including manufacturing, food processing, and pharmaceuticals. It is a cost-effective way to ensure product quality while avoiding the need to inspect every single product in a batch (Selvamuthu & Das, 2018).

There are two main types of acceptance sampling plans: single sampling and double sampling. In single sampling, a single sample is inspected and the decision to accept or reject the batch is made based on the number of defective units in the sample. In double sampling, two samples are inspected. If the number of defective units in the first sample is below a certain threshold, the batch is accepted. If the number of defective units in the first sample is above a certain threshold, the second sample is inspected. If the combined number of defective units in the two samples is below a certain threshold, the batch is accepted. If the combined number of defective units in the two samples is above a certain threshold, the batch is rejected.

The AQL is typically set by the customer and is based on the customer's acceptable level of quality. For example, a customer may set an AQL of 2.5% for major defects, which means that they are willing to accept a lot of products if no more than 2.5% of the products have major defects.

The AQL is used to determine the sample size and acceptance criteria for an acceptance sampling plan. The sample size should be large enough to provide a reasonable level of confidence that the decision to accept or reject the batch is correct. The acceptance criteria should be set such that the probability of accepting a batch that is below the AQL is high, and the probability of rejecting a batch that is above the AQL is low (Guliti, 2018).

2.2 Machine Learning Based Solutions

2.2.1 Types of Machine Learning Algorithms

For any application, developing an effective machine-learning model and tool requires suitable learning algorithms and domain knowledge in the area of the application. According to Lee & Shin (2020) and Ray (2019) machine-learning (ML) algorithms can be categorized into supervised, unsupervised, semi-supervised, and reinforcement learning. According to Schmitt et al.(2020), a fifth category is included in addition to the four categories. The authors defined these categories as follows.

Supervised machine learning is a type of machine learning that learns from labeled data. This means that the data is already tagged with the correct output, so the model can learn how to map input data to output data. This type of machine learning can be used for classification or prediction purposes.

For classification, the model learns to assign input data to one of a set of categories. For example, a model could be trained to classify images of animals into the categories "dog," "cat," "bird," and "other." For prediction, the model learns to predict a value based on input data. For example, a model could be trained to predict the price of a house based on its features.

The labeling of input and output data can be time-consuming and laborious, especially when big data are used. However, supervised machine learning can be very effective in learning complex relationships between input and output data.

Some of the most popular supervised machine learning algorithms include (Lee & Shin, 2020) (Schmitt et al., 2020):

- K-nearest neighbors: This algorithm classifies new data points based on the k most similar data points in the training set.

- Naive Bayes classifiers: This algorithm classifies new data points based on the probability of each class.
- Decision trees: This algorithm classifies new data points by recursively splitting the data set based on the values of the features.
- Linear regression: This algorithm predicts a continuous value based on the values of the features.
- Logistic regression: This algorithm predicts a binary value (e.g., yes or no) based on the values of the features.
- Support-vector machines: This algorithm classifies new data points by finding the hyperplane that best separates the data points into two classes.
- Random forests: This algorithm is an ensemble of decision trees.
- Artificial neural networks: This algorithm is a model that is inspired by the human brain.

Unsupervised machine learning is intended to capture the relationship or correlation among input data for theme analysis or grouping purposes when no information about desired outputs is available. Since there are no output classes that can relate to input data, unsupervised machine learning algorithms attempt to identify similarities between the elements in the input data set and group the elements to gain meaningful insights. Automated multi-document summarization and customer clustering are some of the useful application areas of unsupervised machine learning. Popular unsupervised machine-learning algorithms include clustering, hierarchical clustering, k-means, principal component analysis, and association rules (Lee & Shin, 2020) (Schmitt et al., 2020).

Semi-supervised machine learning is a two-stage or multistage process that uses one learning type as preprocessing for the other, thus employing elements of both. For example, classification tasks via supervised machine learning (e.g., grouping customers into multiple classes) can be performed more accurately and efficiently with the help of seed-training datasets obtained from an unsupervised machine-learning algorithm (e.g., clustering) when no classification is known beforehand, or when labeling of output data would otherwise be time-consuming and costly. In another situation, where only a small portion of the data are labeled because of time constraints, the labeled data can be used to obtain classification rules with supervised machine learning, and these rules can then be used to label the remaining data by unsupervised machine learning. Finally, all labeled data can be used for training with supervised machine learning (Lee & Shin, 2020) (Schmitt et al., 2020).

Reinforcement learning is used to train a software agent how to behave on the basis of environmental feedback and reward mechanisms. While supervised machine learning uses a set of input and output data pairs for training, reinforcement learning tests different actions to determine which ones maximize cumulative reward in an environment, as opposed to simply being told which actions to take. Some popular reinforcement-learning algorithms include Q-learning, hierarchical reinforcement-learning algorithms, temporal-difference learning, and policy-gradient algorithms (Lee & Shin, 2020) (Schmitt et al., 2020).

Active learning aims at finding useful rather than merely statistical findings. Thereby, instead of using statistical evaluations, the supervising user is asked to provide feedback on a question from which the algorithm should learn in a targeted manner (Schmitt et al., 2020).

2.2.2 Image Recognition

Image recognition is an important research direction in the field of image where scientists aim to develop methods to give computers the ability to observe and understand digital images (Yavari, 2020), (Du, 2021). In image recognition, the aim is to replicate the human visual system in computers. Although researchers have progressed significantly in this field, they are still far away from a complete solution (Yavari, 2020).

Despite all the difficulties and limitations of image recognition competence compared to the human visual system, there is still a wide range of applications where image recognition algorithms are used. Character recognition where the computer can automatically read the text, medical diagnosis where image recognition models provide information about patients, face recognition, defect detection in product quality inspection are some of the application areas to mention few (Babic et al., 2021), (Miles & Surgenor, 2011), (Wang et al., 2019), (Yavari, 2020). Moreover, automating the visual inspection for quality checks and fault detection with image recognition has been used in many different areas.

Image recognition algorithms learn different patterns to be able to detect or classify objects. The successful algorithms require large-scale data which is manually annotated. If there is enough data available, there are very effective algorithms that can learn the various types of patterns that characterize an object. The steps included in image recognition process include input data collection in the form of image, image preprocessing, feature extraction from each image and remembered by the deep learning algorithm, and machine learning model that predict and classify objects (L. Li, 2020), (Yavari, 2020).

2.2.3 Deep Learning

Lecun, Bengio, and Hinton (2015) defined deep learning methods as representation-learning methods with multiple levels of representation, obtained by composing simple but non-linear modules that each transform the representation at one level (starting with the raw input) into a representation at a higher, slightly more abstract level. With the composition of enough such transformations, very complex functions can be learned. For classification tasks, higher layers of representation amplify aspects of the input that are important for discrimination and suppress irrelevant variations. Yavari (2020) defined deep learning as a subset of machine learning, which focuses particularly on a type of model known as a deep Artificial Neural Network (ANN). ANNs are inspired and developed by the characteristics of the human brain, but they do not reach the complexity of a human brain. Like that of a biological brain network, artificial networks consist of simple computational neurons that are highly interconnected. The connection of these neurons decides the functionality of the network.

The neuron, also known as a node, perceptron, or unit, is the fundamental building block of an ANN. A neuron can receive a variety of signals as input, either from other neurons or from the outside world. Additionally, relative weights are added to each input to help assess the influence of the related input signal. The weighted accumulation of all the input signals constitutes the output signal of a neuron, which corresponds to the computation in a biological neuron. A complex learning platform is offered when numerous layers of these basic neurons are stacked together and connected to one another. Moreover, when there are many layers of neurons in an ANN then it is often called deep learning where the word deep refers to the depth of the network (Yavari, 2020).

2.2.4 Convolutional Neural Networks (CNN)

CNN is a type of deep neural network which is commonly used for analyzing visual images. It is specifically designed for processing data that has a grid-like structure. CNNs are designed to process data that come in the form of multiple arrays, for example a color image composed of three two dimensional (2D) arrays containing pixel intensities in the three-color channels. CNNs are composed of a series of convolutional layers, pooling layers, and fully connected layers.

The convolutional layers are the most important part of a CNN. The neurons in these layers are only connected to a portion of the neurons in the previous layer. These convolutional layers are responsible for extracting features from the input image. They use small filters to scan the image and extract features. The filters are typically 3x3 or 5x5, and they are applied to the image in a sliding window fashion. The output of the convolution operation is a feature map, which is a representation of the image that is filtered to highlight certain features. Having these locally connected layers makes the network's parameters and computational cost substantially smaller (Yavari, 2020).

The pooling layers are used to down sample the feature maps. This is done by taking the maximum value in each window of the feature map. The purpose of max pooling is to reduce the computational complexity of the network and to help to prevent overfitting.

The fully connected layers are used to classify the image. They are a type of neural network layer that connects all of the neurons in the previous layer to all of the neurons in the next layer. This allows the network to learn complex relationships between the features in the image and the class labels.

The principle of CNN is to use the spatial correlation of the input data to learn features that are relevant to the task at hand. This makes CNNs well-suited for tasks such as image classification, object detection, and segmentation.

There are four key ideas behind CNNs that take advantage of the properties of natural signals: local connections, shared weights, pooling and the use of many layers (Lecun et al., 2015).

There are many different machine learning models that are classified under CNN class. These include VGG, ResNet, YOLO and many more. The best model for a given application will depend on the specific problems at hand and the size of the available dataset.

VGG

VGG (Visual Geometry Group) is a convolutional neural network (CNN) architecture that was developed by the Visual Geometry Group at the University of Oxford. VGG is a very deep CNN, with 16 or 19 layers. This depth helps the architecture to learn more complex features from the data. The architecture of VGG is also based on a stack of convolutional layers, each of which has a small (3x3) filters. This makes the architecture more efficient, and it also helps to prevent overfitting to some level. It can be used for transfer learning. This means that the model can be trained on a large dataset and then fine-tuned for a specific task.

There are different versions of VGG including VGG16, VGG19, VGG16BN, VGG19BN, and VGG-Face. VGG16: VGG16 is a 16-layer CNN architecture. It is one of the most popular VGG models. The VGG16 architecture is based on a stack of convolutional layers followed by fully connected layers. The convolutional layers use small 3x3 filters, which allows the network to learn more local features. The max pooling layers down sample the image, which helps to reduce the computational complexity of the network. The fully connected layers are used to classify the image.

ResNet

ResNet stands for Residual Neural Network, and it is a type of convolutional neural network (CNN) that was developed by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun in 2015. ResNets are known for their accuracy and performance. They are deep CNNs, with hundreds or even thousands of layers. This makes them well-suited for tasks such as image classification and object detection.

The key idea behind ResNets is the use of residual connections. Residual connections are shortcuts that allow the output of a layer to be added directly to the input of the next layer. This helps to prevent the vanishing gradient problem, which is a problem that occurs when the gradients of a deep neural network become too small to be useful. This allows them to learn long-range dependencies in data.

The most popular ResNet variants include ResNet-34, ResNet-50, ResNet-101, and ResNet-152. The numbers represent the number of layers of the network. As the number of layers increases the accuracy of the model increases while the speed is decreasing. There are smaller versions of ResNet. One of them is ResNet-18. It is based on the same residual learning framework as other ResNet variants. It uses residual connections to prevent the vanishing gradient problem, and it has been shown to be effective in a variety of tasks. It is a good choice for tasks where speed and accuracy are both important. The choice of which ResNet version to use depends on the specific task at hand.

YOLO

YOLO is a class of CNN. It stands for You Only Look Once, and it is a type of object detection algorithm that uses a single convolutional neural network to predict bounding boxes and class probabilities for multiple objects in an image. It is also relatively easy to train and deploy, making it a popular choice for many developers.

The YOLO algorithm divides the input image into a grid, and then predicts bounding boxes and class probabilities for each grid cell. The bounding boxes are predicted using a regression model, and the class probabilities are predicted using a softmax classifier.

A softmax classifier is a type of classifier that is used for multi-class classification problems. Multi-class classification problems are problems where there are more than two possible classes that an input can be classified into.

The softmax classifier works by taking the output of a neural network and converting it into a probability distribution over the different classes. The probability distribution is a vector of numbers, where each number represents the probability that the input belongs to a particular class. The number with the highest value in the vector is the class that the input is most likely to belong to.

The YOLO algorithm is a single-shot detector, which means that it predicts all of the bounding boxes and class probabilities for an image in a single pass. This makes YOLO very efficient, as it does not need to iterate over the image multiple times to detect objects.

YOLOv8 is a real-time object detection model that was developed by Ultralytics. It is the latest version of the YOLO object detection model, and it builds on the success of previous versions. YOLOv8 is designed to be fast, accurate, and easy to use.

2.2.5 Recurrent Neural Networks (RNNs)

Recurrent neural network (RNN) is a type of artificial neural network that is well-suited for processing sequential data. RNNs can learn to remember information from previous inputs, which allows them to model temporal dynamics. This makes them a powerful tool for a wide range of tasks, such as sound event detection, speech recognition, machine translation, and natural language processing. The key features of RNNs include recurrent connections, hidden state, and backpropagation through time (BPTT).

RNNs have connections between nodes that create a cycle, allowing output from some nodes to affect subsequent input to the same nodes. This allows RNNs to exhibit temporal dynamic behavior. They have also a hidden state that stores information about the previous inputs. This hidden state is used to compute the output for the current input. They are trained using backpropagation through time (BPTT). BPTT is a variant of backpropagation that is used to train RNNs. BPTT works by propagating the error signal through the network in time.

2.2.6 Transfer Learning

Humans use their past knowledge for learning new objects. The past experience and knowledge help humans to simplify the problem of learning by identifying the old information and learning only the new knowledge (Yavari, 2020). Transfer learning is used to improve a learner from one domain by transferring information from a related domain (Weiss et al., 2016). It is used to implement machine learning algorithms that can transfer knowledge learned from one task to improve learning in a related task (Hussain et al., 2019). These algorithms are particularly important when available training data for a specific task is limited (Yavari, 2020).

Yang, Zhang, Lv, and Wang (2021) proposed an image recognition model with transfer learning for the automatic extraction of image features and the accurate and efficient detection of wind turbine blade damage. Weiss, Khoshgoftaar and Wang (2016) presented a survey of transfer learning methods and application examples. Pan, Pang, Wang, Wang and Chen (Pan et al., 2020) proposed a transfer learning based model for welding defects detection. According to the authors, the experiments of welding defects classification using the dataset verify that the model can accurately identify specific defects on a limited number of training samples. They have carried out a number of experiments for various size images using different transfer learning models. They claimed that the proposed method has better recognition accuracy with smaller model size and less calculation time.

2.3 Related Works

Lei et al.(2018) and Li et al.(2020) have done studies in the area of machine learning driven mobile phone screen defect detection. The traditional way to detect mobile phone screen defects is to arrange inspectors on the production line, where they use the naked eyes to detect the existence of defects on the phone screen in order. The authors of these papers mentioned the drawbacks of the manual detection of mobile phone screen defects.

In the former paper, a high-resolution industrial camera is used to collect mobile phone screen images (Lei et al., 2018). They have collected 3000 image datasets in various product batches and screen models, where each defect in an image is annotated with a bounding box by experienced inspector. To obtain the screen images, they have applied an affine transformation to eliminate lens distortion. They then crop from these images with random positions to obtain multiple screen image patches with the size of 128 × 128. The patches with an overlap larger than 80% with defects are labeled as negative samples (or defect samples). The others are labeled as positive samples. After all, they have produced a total of 27250 positive and 25720 negative samples to form the whole dataset. In this work, the dataset is split into training and testing sets with the numbers of 40000 and 12970, respectively, with 20000 positive/negative samples for training and the rest for testing. Based on the results obtained, they have proposed an end-to-end automated framework integrated with two deep networks for mobile phone screen defect detection. The first network aims at parallel detecting defects under multiple scales. The second one is designed for predicting the area with the most probability of containing defects and eliminating the environmental factors on capturing.

In the later paper, the authors proposed a defect extraction and classification scheme for mobile phone screen based on machine vision (C. Li et al., 2020). The main goal of this work is to extract and recognize some types of typical mobile screen defects according to the defect detection standards with high accuracy and efficiency. In order to improve the efficiency of the algorithm, they have designed a pre-examination algorithm and a coarse precise defect extraction strategy. According to the paper, the mobile phone screen can be divided into several parts. It contains camera hole, receiver hole, IR hole, edge, printed area, window area and logo. The types of defects in each part are different, and their corresponding images are also very different. Therefore, it is needed to extract the regions of interest (ROIs) to avoid the interference between different parts. The design of the mobile phone screens is also changing fast, and there have already been a variety of screen styles. Therefore, the process of obtaining ROIs needs to consider the generality of the algorithm. To solve these problems, the authors proposed a ROI acquisition method based on template matching technology. In this work, the detection criteria are defined, and a classification algorithm combining multi-layer perceptron (MLP) and deep learning technologies is proposed. Based on the experimental results, the authors claimed that satisfactory performance is achieved in detecting scratches, floaters, light stains and dark stains of the mobile phone screen with the proposed detection scheme.

S S Zakaria1, A Amir & N Yaakob1 (2020) have studied the contribution of machine learning based technologies for detection of defects in printed circuit board (PCB) assembly line. They have reviewed various types of PCB defect detection methods and it has compared the available machine learning based PCB defect detection approaches and other PCB defect detection methods that do not apply machine learning approaches. They have concluded that the application of machine learning for PCB defect detection is at the infant stage.

Sun, Wang, Wang, Li, Li , and Fu (2021) studied the application of FSL for remote sensing and object detection. They provided a bibliometric analysis of the existing works for remote sensing interpretation related to FSL. Sung, Yang, and Zhang (2018) proposed a FSL method called Relation Network (RN) for image recognition. They proposed a two-branch RN that performs few-shot recognition by learning to compare query images against few-shot labeled sample images.

2.4 Summary of Literature Findings and Gaps Identified

From the discussions presented above, one can conclude that developing solutions for quality inspection problems through machine learning based approaches is an active research area at the current time. The main literature findings include:

- The theoretical background in the area of ML and quality inspection.
- ML model development procedures.
- The importance of having a good understanding of the inspection problem before developing an ML model.
- The need to collect a large and representative dataset of good and defective components and devices.
- The importance of choosing the right ML algorithm for the specific inspection problem.
- The need to evaluate the performance of the ML model on a representative test dataset.
- To tackle data shortage, data augmentation and transfer learning are the available solutions that can be used.

Transfer learning is used to improve a learner from one domain by transferring information from a related domain. But, most of the successful deep learning algorithms need large-scale data and each algorithm can basically master only one task. As a result, transferring information from successful model in a related domain to the domain at hand may not be easy. The various solutions proposed to tackle this issue are at their infant stage.

Chapter 3 Methodology

The main phases of the methodology used in this thesis work are discussed as follows.

3.1 Literature Review

In this thesis work, state-of-the-art literatures in the area of quality inspection in cell phone and electronics manufacturing and assembling industries and in the area of machine learning algorithms and models for quality inspection are reviewed. A thorough understanding of the theoretical background of the thesis is acquired through this part of the thesis work. These acquired knowledge and skills are utilized in the other stages. The summary of the review and the background theory of the thesis work is presented in chapter two.

3.2 Data Collection

In the cell phone assembly process of the company, there are more than six inspection points. The data regarding to the parameters to be inspected, the inspection procedures, the steps of inspection, the defect types, and the conditions to be fulfilled in performing the inspection are collected. In order to fully understand the nature of the defect types in the inspection points, sample pictures of defective components are taken. Monthly and yearly inspection reports are also collected.

For the development of machine learning based inspection model for the cell phone surface defect detection, pictures of defective and defect free phone screens are taken. In addition to this, the experts who are directly engaged with quality inspection and direct observation are interviewed. The focus of the interview was understanding the inspection process and the challenges encountered in performing the inspection. In addition to these, secondary data are collected from literatures for transfer learning.

3.3 Studying the Natures of the Problems for Each Inspection Point

Based on the collected data and the identified tools and algorithms in the literature review part, the nature of the problems and the parameters for every inspection point in the assembly process are studied. In this stage, based on the nature of the problems, literature review and machine learning expert opinion, the problems to be solved by machine learning driven approaches are identified.

3.4 Select Algorithms for the Specific Cases Based on the Nature of the Problems

The problems (inspection tasks) that are to be solved and improved by machine learning based solutions are considered in this stage. Based on literature, expert opinion and the nature of the problems, selection of machine learning algorithms and models are done. In this stage, the inspection problems in the assembly line that are identified to be solved by machine learning driven approaches are categorized into two based on their nature. These are visual inspection problems that can be solved by image processing solutions and sound inspection problems that can be solved by sound processing solutions. For the image based visual inspection problems, CNN class algorithms are proposed. Recursive neural network and CNN class algorithms are proposed for that of the sound inspection problems.

3.5 Demonstration through Detailed Model Development for One Inspection Problem

In this stage, one inspection problem is selected and a detailed machine learning driven model is developed for demonstrating the advantages and improvement capability of machine learning for quality inspection problems. The following methodology is used at this stage.

a) Selection of inspection problem in the assembly process

In this stage, one inspection problem is selected from the receiving inspection sensitive and valuable components for detailed model development and demonstration. It is selected from the receiving inspection because of three reasons. These are

- The receiving inspection is the first stage of the inspection process. Improving the efficiency at the early stage of the inspection results in the reduction of the cost and time spent in the assembly process of a product.
- At the receiving inspection, the printed circuit board assembly (PCBA), 3 in 1 (up housing, touch panel (TP) and LCD), camera and battery are the most sensitive components and the cost implication of the occurrence of defect in these components is high. As a result, 100% inspection (no sampling) for the whole purchased components is performed. This means the inspection volume of these components is very high.
- The availability of data for training the machine learning algorithm.

b) Data set preparation

Because there are no historical data available, pictures of defective and defect free components are taken for the specific inspection problem. As will be discussed later, due to the shortage of defective components for the selected problem, a dataset is collected from other source. After collecting the data, annotating and labelling of the data, class balancing of the data, partitioning the data into training and validation/testing sets as well as some preliminary data pre-processing is done to make the dataset complete and usable for machine learning algorithms.

Selection of algorithm for the selected problem

Based on literature, the nature of the specific problem and expert opinion, three different CNN algorithms are selected and comparison of these algorithms is made using machine learning performance metrics such as accuracy, speed. These algorithms are Residual Network (ResNet), You Only Look Once (YOLO) and Visual Geometry Group (VGG).

c) Training of the model

Using the portion of the data set for training, the selected machine learning models are trained. The learning curve of the training process is prepared by making a validation test in some intervals of the training epochs for each model. Since the number of dataset samples are smaller due to shortage of data, the same testing and validations set are used.

d) Testing of the model

The test dataset split is used at this stage for testing the performance of the trained models. The test results have showed the performance of each model.

Draw conclusions

Based on the results, conclusions and recommendations are drawn.

Chapter 4 Results and Discussion

4.1 Current Situation of the Quality Inspection Process

4.1.1 The Inspection Process

In order to fully understand the existing quality inspection process, the full process of the cell phone assembly is visited and the inspection personnel are interviewed. The interviews are focused on the parameters to be inspected, the defect types, the inspection processes, the standards used and the challenges encountered in performing the inspection duties. During the visit, it is observed that the list of the parameters to be inspected, the defect types, the steps of inspection and the conditions to be fulfilled in performing the inspection with checklists are given in printed form for every inspector at all the inspection points.

The inspection process in the smart phone assembly line is classified into three as receiving inspection, in-process inspection and final inspection. The defect levels are also classified into three as critical, major and minor defects. Critical defects are defects that pose threat to the user's personal or property safety. Defective products are classified in this class if they do not conform to the national related laws and regulations and relevant standards. When the functions of the product or its appearance cause consumer dissatisfaction, it is categorized as major defect. Whereas, minor defect is a defect type that affect the degree of consumer satisfaction.

The current situation of the quality inspection process is summarized and presented as follows based on the visit and the interviews.

Receiving Inspection

Receiving inspection is an acceptance activity oriented to confirm that the components received are in accordance with the requirements set. It validates the quality of purchased raw materials based on set acceptance criteria. There are four inspection steps in this class and there are a number of components and parameters to be inspected at each step.

The first step is label inspection. It is dedicated to the inspection of the label on the packaging box of components. The parameters to be checked include model number, product code, software version, material description, material quantity, and supplier. These parameters are checked according to the agreed criteria. A sample packaging box label with the inspected parameters is shown in Figure 4.1.

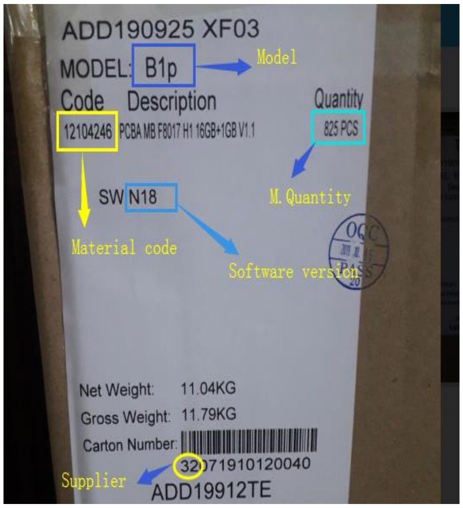

Figure 4.1: Sample Packaging Box Label with Inspected Parameters Indicated

The second step is to inspect the physical condition of the packaging box and the packaged materials. These include to check and inspect whether the box and the materials are neat, whether there is bruise, crushing, or any other physical damage.

The third step is to check the materials according to AQL standard. As discussed in chapter two, AQL is a method used to define lowest or worst quality level and a production order to find whether or not the entire product order has met the client's specifications based on sample size.

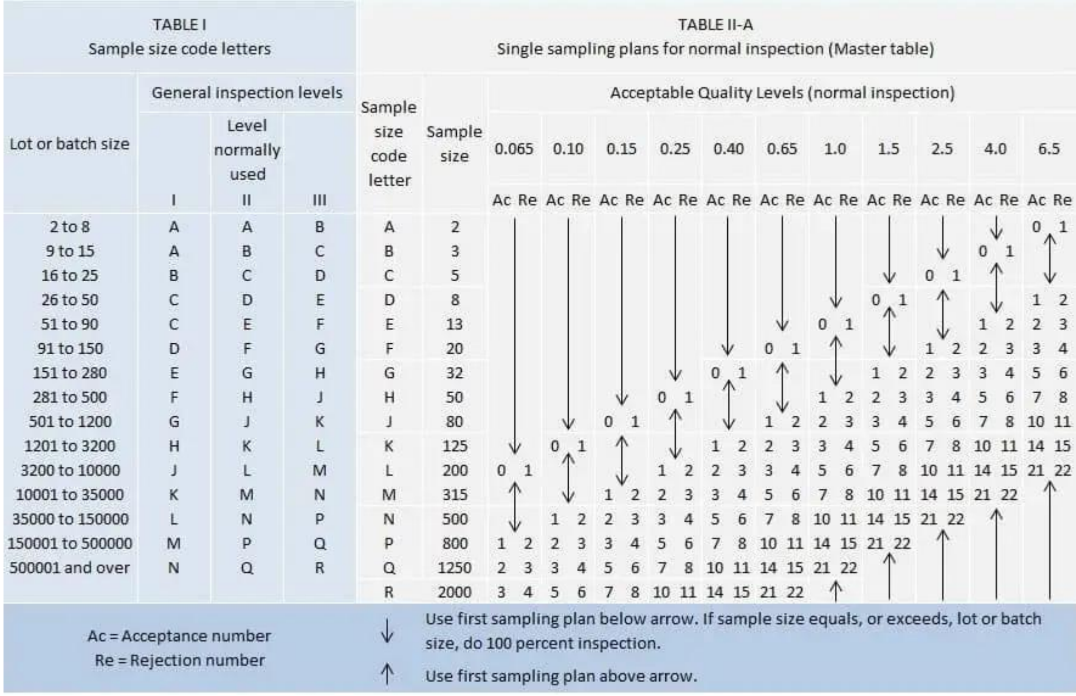

TABLE I Sample size code letters				TABLE II-A Single sampling plans for normal inspection (Master table)																							
Lot or batch size	General inspection levels			Sample size code letter	Sample size	Acceptable Quality Levels (normal inspection)																					
	I	Level normally used II	III			0.065		0.10		0.15		0.25		0.40		0.65		1.0		1.5		2.5		4.0		6.5	
						Ac	Re	Ac	Re	Ac	Re	Ac	Re	Ac	Re	Ac	Re	Ac	Re	Ac	Re	Ac	Re	Ac	Re	Ac	Re
2 to 8	A	A	B	A	2	↓		↓		↓		↓		↓		↓		↓		↓		↓		↓		0	1
9 to 15	A	B	C	B	3	↓		↓		↓		↓		↓		↓		↓		↓		↓		0	1	↑	
16 to 25	B	C	D	C	5	↓		↓		↓		↓		↓		↓		↓		↓		0	1	↑		↓	
26 to 50	C	D	E	D	8	↓		↓		↓		↓		↓		↓		↓		0	1	↑		↓		1	2
51 to 90	C	E	F	E	13	↓		↓		↓		↓		↓		↓		0	1	↑		↓		1	2	2	3
91 to 150	D	F	G	F	20	↓		↓		↓		↓		↓		0	1	↑		↓		1	2	2	3	3	4
151 to 280	E	G	H	G	32	↓		↓		↓		↓		0	1	↑		↓		1	2	2	3	3	4	5	6
281 to 500	F	H	J	H	50	↓		↓		↓		0	1	↑		↓		1	2	2	3	3	4	5	6	7	8
501 to 1200	G	J	K	J	80	↓		↓		0	1	↑		↓		1	2	2	3	3	4	5	6	7	8	10	11
1201 to 3200	H	K	L	K	125	↓		0	1	↑		↓		1	2	2	3	3	4	5	6	7	8	10	11	14	15
3200 to 10000	J	L	M	L	200	0	1	↑		↓		1	2	2	3	3	4	5	6	7	8	10	11	14	15	21	22
10001 to 35000	K	M	N	M	315	↑		↓		1	2	2	3	3	4	5	6	7	8	10	11	14	15	21	22	↑	
35000 to 150000	L	N	P	N	500	↓		1	2	2	3	3	4	5	6	7	8	10	11	14	15	21	22	↑		↑	
150001 to 500000	M	P	Q	P	800	1	2	2	3	3	4	5	6	7	8	10	11	14	15	21	22	↑		↑		↑	
500001 and over	N	Q	R	Q	1250	2	3	3	4	5	6	7	8	10	11	14	15	21	22	↑		↑		↑		↑	
				R	2000	3	4	5	6	7	8	10	11	14	15	21	22	↑		↑		↑		↑		↑	

Ac = Acceptance number
Re = Rejection number

↓ Use first sampling plan below arrow. If sample size equals, or exceeds, lot or batch size, do 100 percent inspection.

↑ Use first sampling plan above arrow.

Figure 4.2: AQL Table Adopted in the Company

The fourth step is visual inspection. This step includes the visual inspection of the printed circuit board assembly (PCBA), 3 in 1 (up housing, touch panel (TP) and LCD), camera and battery. The PCBA is checked for short circuit, component pad lost, software version mix, battery and USB connector broken, component lost, reverse IC or reverse displacement, rust, code mix, and sim holder broken. A sample picture of PCBA is shown in Figure 4.3 below.

There are different parameters and components to be inspected at the front, corner and back sides of the 3 in 1. The pictures of front and back sides of a 3 in 1 are shown in Figure 4.4. In the front side, the main LCD display or glass and the TP are checked for broken glass, broken TP, dirt, abnormal color dot, scratch, and other abnormal appearances. In the corner side, the up housing is checked for whether the material is broken or not. The back side of the 3 in 1 is inspected for broken flexible printed circuit (FPC) pin of the LCD, whether the LCD's FPC pin pad is lost or broken, whether the TP's FPC pin is lost, broken or short.

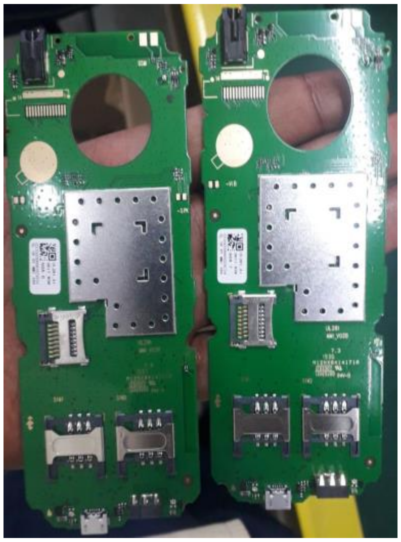

Figure 4.3: Sample Printed Circuit Board Assembly

The next component to be visually inspected is the phone camera. At the front side, the camera lens is checked for gum, damage, bad dirt appearance, crack, and so on. The camera sit at the end and steel connections are checked for tainted gum, damage, crack, steel rust, scratches, oxidation and other undesired phenomena.

The other component is the battery. First the battery brand, model, and specification are checked for correctness. Then, it will be checked for deformation, expansion and other undesirable phenomena. The battery terminals are inspected for oxidation, black, scratches, whether the polarity labels are correct or not, and other abnormalities.

Figure 4.4: Front and Back Sides of a Sample 3 in 1

In-process Inspection

It is an in-process quality check for the intermediate and finished products during the assembly process. There are 23 inspection steps performed during this stage. It includes the appearance and functional quality tests during the assembly process and after the smart phones are fully assembled. The appearance inspections are performed to check for appearance defects that occur during assembly and missed from receiving inspection. The types of appearance defects in this inspection stage are listed below.

- Straight line or curve (scratch, moulage）
- Dot defect (splodge, dust, colorful dot, bright dot, dent etc.）
- Assembly gap (keypad, lens, up & low housing, battery cover)
- Suitability (battery cover, SIM, battery, battery connection pins, etc.)
- Assembly defects (screw, screw cover, malposition deviation)
- Label (include LOGO) miss, misprint, LOGO not clear

The functionality tests are performed to check the correct functionality of the components, sub systems, and the whole smartphone. The functional components and systems tests include battery connector test, camera test, charging test, software version test, backlight test, color display test, button test, touch screen test, Wi-Fi test, Bluetooth test, storage test, speaker test, microphone test, receiver test, headphone test, FM test, sim card test, acceleration test, brightness adjustment test, flash test distance sensing test and aging test etc.

Final Inspection

It is the inspection performed in the final stage of the manufacturing process. It concentrates in the quality check of the final packaging of the products. In this inspection stage the weight of the packaged smartphone with its accessories is measured to check whether all accessories are included in the packaging box or not. The packaging box serial number and barcodes are also checked. In addition to that, most of the in-process inspection functional tests are repeated here by taking randomly sampled phones. These include battery connector test, camera test, charging test, backlight test, color display test, button test, touch screen test, Wi-Fi test, Bluetooth test, storage test, speaker test, microphone test, sim card test, acceleration test, brightness adjustment test and flash test.

4.1.2 Existing Challenges in the Inspection Process

The main challenges observed in the inspection process in the case company are listed as follows.

- As mentioned before, all the components should be inspected without sampling for the PCBA, the 3 in 1, the camera and the battery in the receiving inspection. For the inspection of these components, the inspection volume is very high. Due to various reasons such as inspector lack of attention, skill gap, rush to avoid manufacturing delay, and subtle nature of defects, defective components are missed detection. From the 2022 one year inspection report, it is found that from the total defective components detected at the final inspection, more than 31% of the defective components are missed undetected from the receiving inspection. 64% of these missed defects resulted in the total rejection of the assembled phones.
- Most of the inspection techniques used in the company are manual based and they take longer time of inspection. For example, by a well-trained inspector, it takes at least three seconds to inspect surface defects of a single phone (3 in 1) screen. For a single cell phone, there are at least 26 components to be inspected. In addition to the time required for inspection, there is a time gap between consecutive inspections. This results in a manufacturing delay.

4.2 ML Based Solutions for the Quality Inspection Problems

The detailed discussion of the inspection process and the natures of the inspection problems in the company is presented in the previous section. In this section, first the categorization of the inspection problems as problems to be solved by machine learning based solutions and problems to be addressed by other methods is presented. Then, for the former problems machine learning based approaches with the ML model development procedure are proposed.

4.2.1 Categorization of the Inspection Problems

The inspection problems in the inspection process of the case company are categorized into five as image based, sound based, other functional tests, weight test and simple code tests (serial number (SN), quick response (QR) code and barcode tests) as shown in Table 4.1.

The image based inspection problems during the receiving inspection are visual inspection problems that require little or no human to component interaction. Whereas, most of the image based inspection problems at the in-process and final inspection require high rate of human to component interaction. For the former image based inspection problems, the deployment of ML based solutions is relatively simpler. For the later inspection problems, the ML model deployment may require advanced robotic system that can interact with the inspected phone during inspection.

The image based and sound based inspection problems are categorized under inspection problems to be addressed by ML based approaches. The reasons for this and the model development procedures are discussed in the next consecutive sections.

The other functional tests, the weight test and the simple code test inspection problems are better suited to other deterministic problems. This is because, the nature of the defect types of these inspection problems are deterministic.

4.2.2 Image Based Inspection Problems

As discussed in chapter three and can be seen in Table 4.1, most of the inspection problems in the company are image based problems. These image based inspection problems are best suited for ML based solutions.

Table 4.1: Categorization of the Inspection Problems

Inspection Class	Problems Addressed by ML Based Approaches		Problems Addressed by OtherApproaches		
	Image Based	**Sound Based**	**Other Functional Tests**	**Weight Test**	**Serial Number (SN), QR Code and Barcode**
Receiving Inspection	Label Inspection Packaging Box Inspection PCBA Inspection 3 in 1 Inspection Camera Inspection Battery Inspection	-	-	-	Packaging Box SN Packaging Box Barcode
In-process Inspection	Battery Connector Test Camera Test Charging Test Software Version Test Backlight Test Color Display Test Button Test Touch Screen Test Wifi Test Bluetooth Test Storage Test	Speaker Test Microphone Test Receiver Test Headphone Test FM Test	Simcard Test Acceleration Test Brightness Adjustment Test Flash Test Distance Sensing Test Aging Test	-	Mother Board SN Mother Board QR Code Aftermarket lebel QR Code
Final Inspection	Battery Connector Test Camera Test Charging Test Software Version Test Backlight Test Color Display Test Button Test Touch Screen Test Wifi Test Bluetooth Test Storage Test Packaging Box Appearance Test	Speaker Test Microphone Test Receiver Test Headphone Test FM Test	Simcard Test Acceleration Test Brightness Adjustment Test Flash Test	Single Packed Phone Weight Test Multiple Phones Package Weight Test	Packaging Box SN Packaging Box Barcode

The main reasons are summarized as follows.

- ML algorithms can learn to identify patterns in images that are difficult for humans to see. This is because ML algorithms can be trained on large datasets of images, which allows them to learn the statistical relationships between different features in images.
- ML algorithms can be very accurate at image classification. This is because they can learn to identify even subtle patterns in images that humans might miss.
- ML algorithms are scalable. They can be used to classify large datasets of images without the need for manual intervention. When the company increases the production capacity, the inspection volume will increase. The developed ML models will easily handle such high volume inspection.

As discussed in chapter two, CNNs are the best types of ML models for image classification and segmentation. Therefore, CNN class algorithms are highly suggested for developing ML based inspection models for these image based inspection problems in the inspection process. The best model for a given application will depend on the specific problems at hand and the size of the available dataset. The procedure used for developing ML based inspection model for these problems is suggested as follows.

1. Prepare a dataset. The first step is to prepare a dataset that is used to train the model for the specific problem. In a given inspection point in the inspection process, when a defect is inspected, its picture has to be taken and a labeled image has to be documented. The labeling and documentation can be simply done by having folders named by the defect type and to collect the images in these folders through time. Datasets from other sources may be used depending on the specific problem at hand. In addition to images of defective components, a class of defect free component images has to be prepared.
2. Preprocess the data. Once the dataset for the specific problem is prepared or chosen, preprocessing of the data is required. The purpose of preprocessing is to convert the data to a format that can be used by the model. This may involve resizing the images, converting them to grayscale, or normalizing the pixel values.

3. Choose a model architecture. There are many different model architectures available, but some popular options for image classification include YOLO, ResNet, and VGG. The choice of model architecture will depend on the specific task that is to be achieved.
4. Train the model. Once the model architecture is chosen, it is required to train the model on the dataset. This can be done using a variety of different frameworks, such as TensorFlow, PyTorch, or Keras.
5. Evaluate the model. Once the model is trained, its performance need to be evaluated on a held-out test set. This will give an idea of how well the model will generalize to new data.
6. Deploy the model. Once the model is evaluated and its performance is good, the next step is to deploy it to production. This may involve making the model available as a web service or integrating it into the application at hand. This may be integrating in a conveyor, robotic or some other automation system. Most of the inspection problems at the receiving inspection can be implemented and integrated in simple conveyor systems. But, the visual based functional tests in the in-process inspection may require advanced robotic systems for ML model deployment.

Here are some additional steps that may be taken depending on the specific task:

Fine-tune the model. If the model is not performing as intended, it may be required to be fine-tuned. This involves adjusting the model's parameters to improve its performance on the specific task.

Augment the data. If the dataset is small, it may be required to augment the data. This involves creating new data by applying transformations to the existing data. This can help to improve the model's performance by making it more robust to variations in the data.

Use transfer learning. If a large dataset is not available, transfer learning can be used. This involves using a pre-trained model as a starting point for the model to be developed. This can help to improve the model's performance by initializing the model with weights that have already been learned on a large dataset.

4.2.3 Sound Based Inspection Problems

ML based sound event detection (SED) models can be used for quality inspection of audio systems and components. SED can be used to detect defects in speakers, microphones, and other audio equipment. ML based approaches are becoming increasingly popular for SED because they have been shown to be more accurate. They require large datasets of labeled audio recordings to train the ML algorithms.

ML based SED models can be used to detect defects in the phone speakers, such as blown speakers or damaged cones. They can also be used to measure the frequency response of speakers, which can be used to assess their overall quality. To detect the defects of mobile phone microphones, such ML based SED models can be used. The defects in the microphone may be damaged diaphragms or poor sensitivity. ML based SED models can also be used to measure the frequency response of microphones, which can be used to assess their overall quality.

The procedures for developing quality inspection ML models for speaker and microphone inspection are summarized as follows.

1. Prepare a dataset: The first step is to collect a dataset of labeled audio recordings. This dataset should contain recordings of different types of mobile phone speaker defects, microphone defects and headphone defects. The recordings should also be labeled with the start and end times of each defect.
2. Preprocessing the data: The next step is to prepare the data for training the machine learning model. This involves preprocessing the data, such as removing noise and normalizing the audio levels. It also involves extracting features from the audio recordings that can be used to identify different defects.
3. Model selection: The next step is to select a machine learning model for training. There are many different machine learning models that can be used for speaker, microphone and headphone inspections, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) as discussed in chapter two. The best model for a given application will depend on the specific defects to be detected and the size of the dataset.
4. Model training: The next step is to train the machine learning model. This involves feeding the prepared data to the model and allowing it to learn to identify different defects.
5. Model evaluation: Once the model is trained, it is important to evaluate its performance. This can be done by testing the model on a test dataset of audio recordings. The evaluation results will show whether the model is accurate enough for the application at hand.

6. Model deployment: Once the model is evaluated and found to be accurate enough, it can be deployed for use. This may involve integrating the model into a software application or hardware device.

4.3 ML Model Development for the Screen Defect Inspection

4.3.1 Selection of Inspection Problem in the Assembly Process

Based on the criteria mentioned in chapter one, the screen defect inspection of the 3 in 1 is selected for detailed ML model development. The big challenge here was the absence and shortage of data suitable for ML model training. Most of the inspection problems at the receiving inspection are image based problems that are inspected by visual means. As discussed in chapter two and section 4.2, for such problems a large amount of image dataset is required for ML based model development.

It was hard to find defective components within the given timeframe of the thesis work in the company. This is because the defective components during and after the assembly process are disassembled or scraped without taking and having pictures with the necessary documentation. For example, there were a large number of defective PCBA components in the company. Because one or more components are disassembled from all the available circuit boards, they are not suitable for ML. The component losses will provide misinformation to the model during training. Datasets of PCBA can be found from some other sources but they are completely different from the PCBA's of the case company. As a result PCBA is not selected for detailed model development.

For the smartphone screen (3 in 1 screen) small number of defective screens are found and their pictures with the defect free screens are taken. For this reason, this inspection problem is selected for model development for demonstration purpose. In addition to this, datasets from other sources can be used due to the similar features of smartphone screens from different manufacturers.

4.3.2 Dataset Preparation

Due to the small number of the available defective smartphone screens in the company, mobile phone screen surface defect segmentation (detection) dataset (MSD) from GitHub, Inc. is used. GitHub is a code hosting platform to build, scale, and deliver secure software. The MSD dataset contains 3 types of surface defects of mobile phone: Oil, Scratch and Stain. It consists of 1200 images and 400 images for each defects. In addition to that, there are 20 good (defect free) phone screen images in the dataset. The dataset is built and presented by Jian Zhang, Miaoju Ban (Zhang, 2022).

One of the purposes of this thesis work is to develop an ML model that can identify defective and defect free phone screens and classify the type of the surface defect. As a result, there should be a class of defect free (good) images of phone screens with the same number of images with that of the other defect classes. In order to have class balance, 380 pictures of phone screens are taken manually and the images are added to the 20 good class images of the MSD dataset.

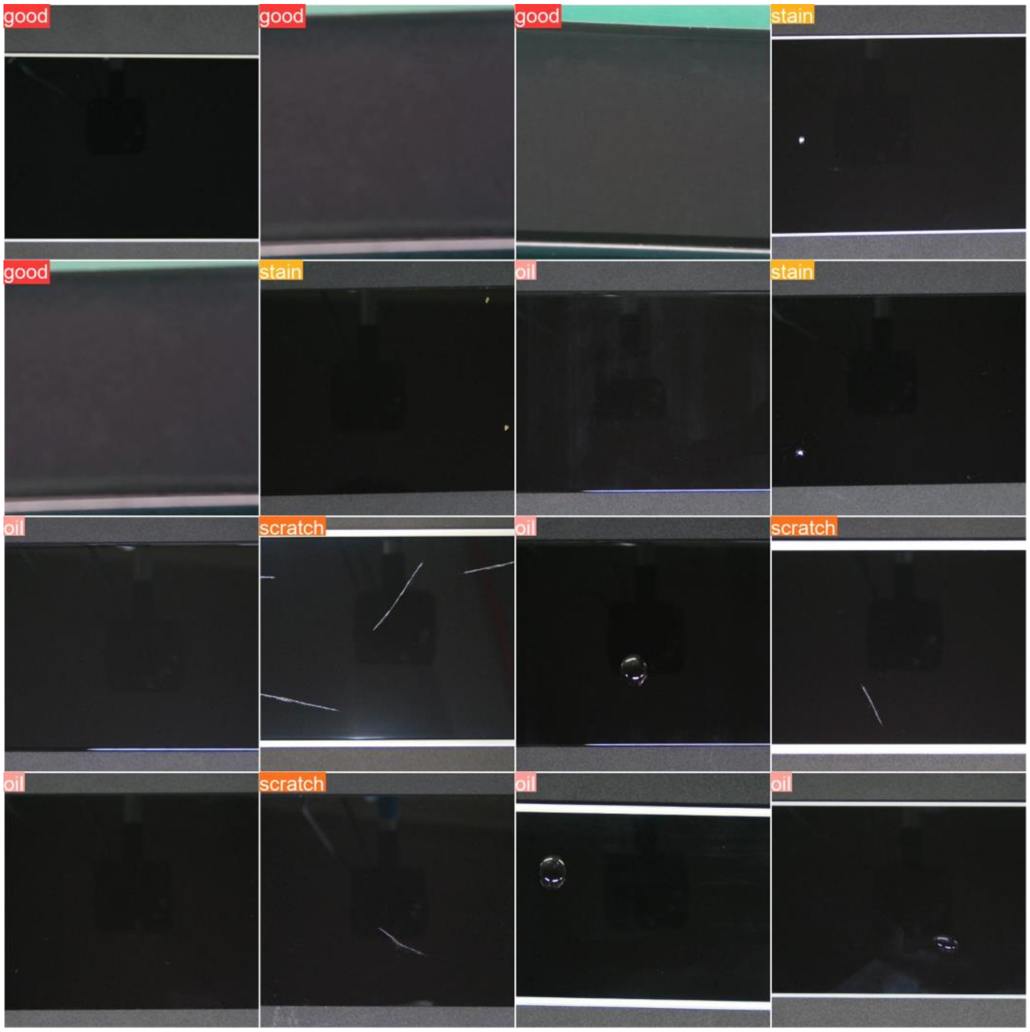

Figure 4.5: Sample Labelled Images of the Four Classes

Because the 1220 images of the MSD dataset are already preprocessed, preprocessing is performed only for the additional 380 pictures. The new dataset consists of 1600 images and 400 images for each class: Oil, Scratch, Stain and Good. The 400 images in each class of the dataset are randomly divided into training and testing with ratio of 8:2. A total of 1280 images are used for training and the rest 320 images are used for testing. The 20% testing images from each class are taken by using random sampling. Sample pictures of the labeled images are shown in Figure 4.5 above.

A second dataset is prepared by using only the pictures taken manually from the company. The dataset consists of 76 images. It is categorized into two classes: Defect and Good. In the Defect class, 25 images are available and the remaining images are for the Good class. Most of the defects in this dataset are broken screens that can easily be identified through human eye. They are taken due to unavailability of reasonable defective screens that show the actual challenge in the inspection process.

This dataset is prepared to demonstrate how a labelled image dataset is prepared from scratch manually by taking pictures of defective and defect free components for ML based image classification. From the two classes, 6 images from the defect class and 8 images from the good class are taken by random sampling for model testing. The remaining images in the two classes are used for training the model. Sample pictures of the labeled images in this dataset are shown in Figure 4.6 below.

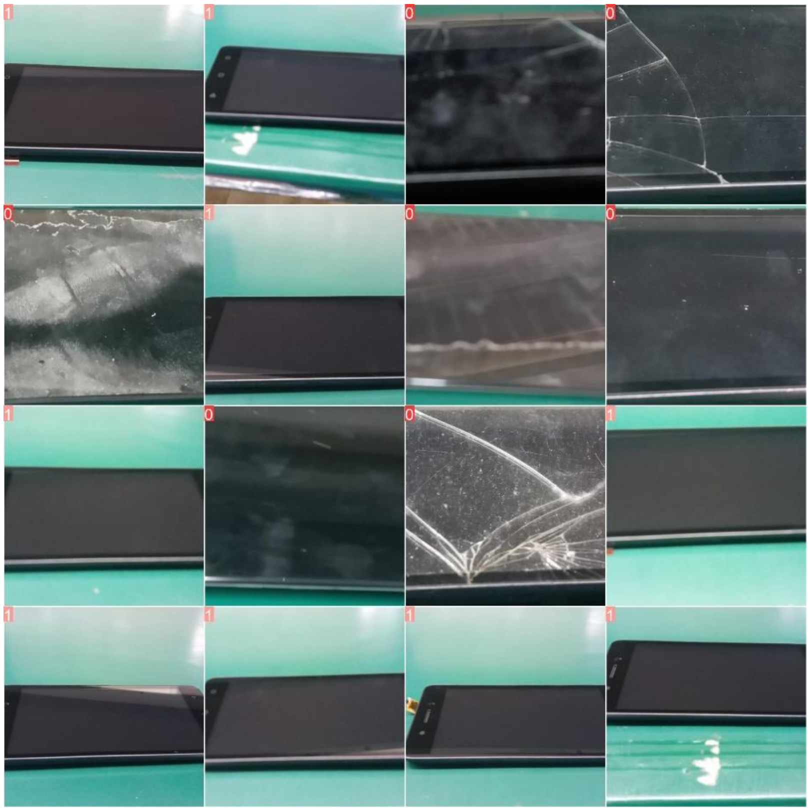

Figure 4.6: Sample Labelled Images of the Two Classes

4.3.3 Selection of Algorithm for the Selected Problem

As discussed in chapter two and four, machine learning algorithms can learn to identify patterns in images that are difficult for humans to see. This is because machine learning algorithms can be trained on large datasets of images, which allows them to learn the statistical relationships between different features in images. Three CNN class ML models are selected for developing the phone screen defect detection ML model. These models are VGG16, ResNet-34 and YOLOv8.

VGG16 is a powerful CNN that has been used for a variety of tasks, including image classification, object detection, and segmentation. It is one of the first CNNs to achieve state-of-the-art results on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). When it is compared with VGG19, the number of layers and parameters are smaller. For faster training and run, it is the better choice.

In this thesis work, PyTorch ML Framework is used for training and model development. In this framework the available pre-trained ResNet model libraries are ResNet-18, ResNet-34 and ResNet-50. For selecting from the three models, training and testing of the models using the dataset with 1600 images is performed. The ResNet-18 model results overfitting whereas the ResNet-50 model results under fitting. As a result, the ResNet-34 model is selected.

YOLOv8 is a powerful object detection model that is well-suited for a variety of tasks. It is a good choice for users who are looking for a fast and accurate model that is easy to use. It is selected because it is the latest optimized version of the YOLO class models.

4.3.4 Training and Testing of the Models

The three models are trained and tested using the training and testing images of the dataset with 1600 images. The training and testing is performed on high performance computer with CPU: MSI Core i7 9th Generation with 32 GB RAM and GPU: NVIDIA GeForce RTX 2080 Ti with 8 GB graphics memory using PyTorch framework. PyTorch is an open source machine learning (ML) framework based on the Python programming language and the Torch library.

First, the three models are trained to build new models from scratch using the mentioned dataset with a training epoch of 50. An epoch is one complete pass through the entire training dataset. This means that the model will see every training image once during an epoch. The number of epochs is a hyper-parameter that determines how many times the model will see the entire training dataset.

The performances of the three developed models are then compared and the two models with better performance are selected for further training. The pre-trained ResNet-34 and YOLOv8 are selected for fine tuning of the parameters and transfer learning. The number of epoch used for training these two models is 30. It is found that the fine-tuned YOLOv8 model performance is better than that of the fine-tuned ResNet-34.

Then, the second dataset with two classes is used for training and testing the pre-trained YOLOv8 model with epoch number of 50. The YOLOv8 model is selected due to its better performance than that of ResNet-34. The test results and the comparisons of the performances of the models are discussed in the next section.

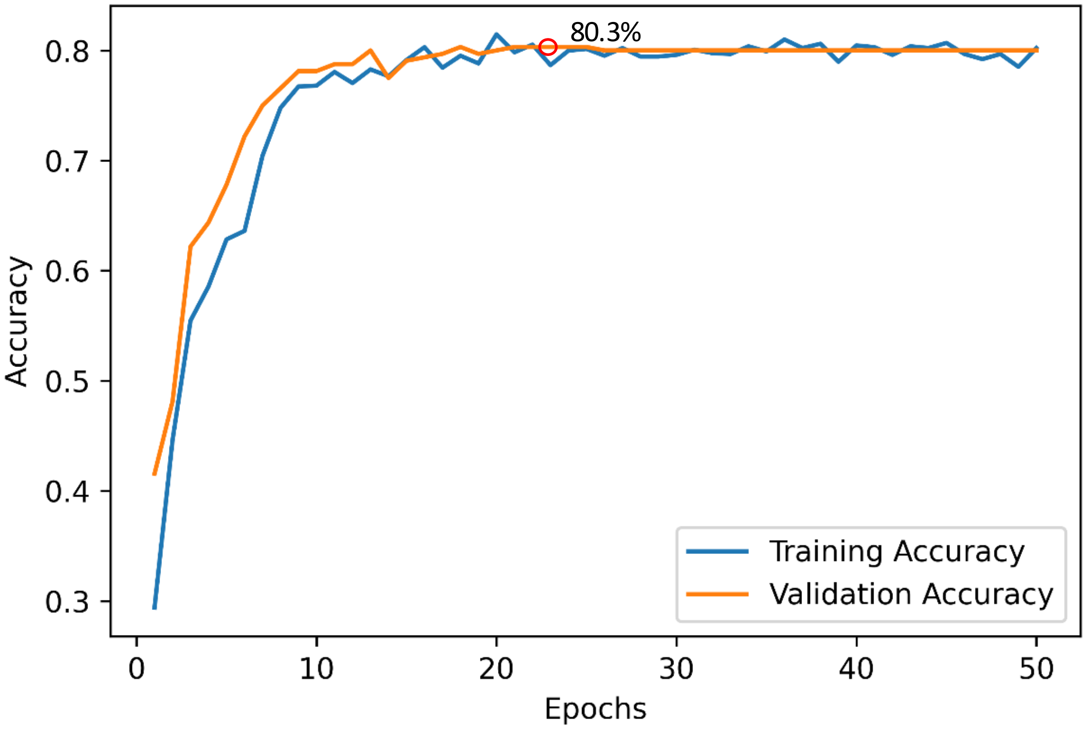

Figure 4.7: Training and Validation Accuracies of VGG16 from Scratch Model

As discussed previously section, five comparative experiments are performed with the three ML models using the first dataset. Then, one experiment is performed with YOLOv8 using the second dataset. The training and validation results with the performance comparisons are summarized as follows.

VGG16 Training and Testing Results

VGG16 model is trained using the 1280 training images of the first dataset to build a new model from scratch. The remaining 320 images are used to test the model. The training and validation accuracies during the training process are shown in Figure 4.7. The training and validation losses are also shown in Figure 4.8.

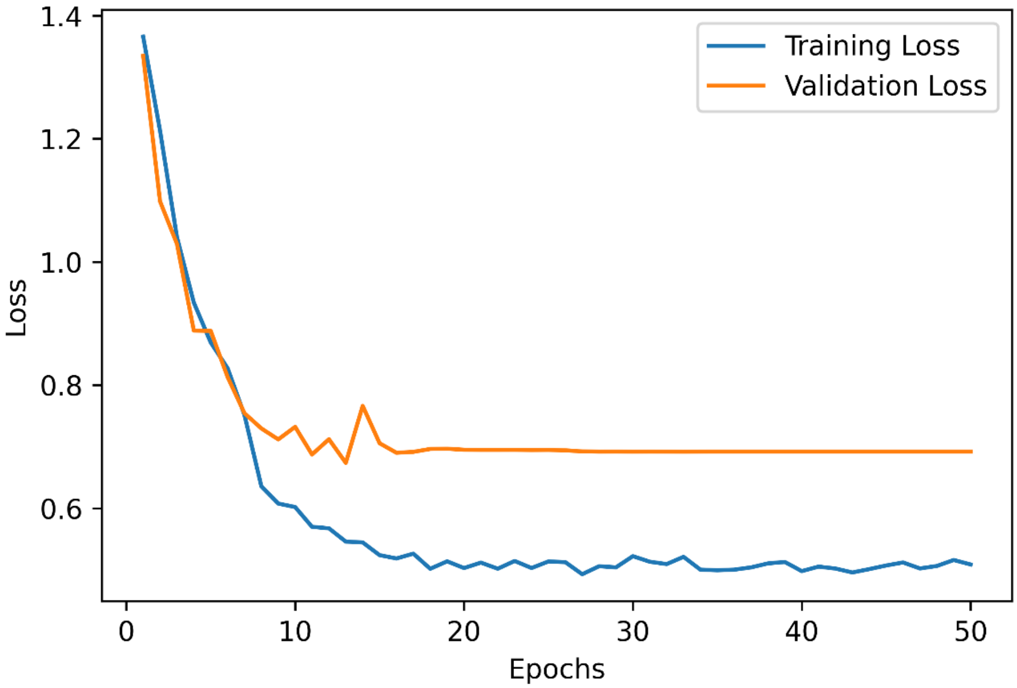

Figure 4.8: Training and Validation Losses of VGG16 from Scratch Model

The purposes of the training accuracy and loss are to show the progress of the training process. Based on these values at each epoch, the model updates its trainable parameters to have better performance.

The performance of the model at every epoch is validated by using the test data and the validation accuracy and loss are generated. These values show the performance of the model at that epoch. They do not affect the training process. This means, the trainable parameters are not updated based on these validation accuracy and loss.

The validation accuracy is calculated as the ratio of the summation of the correctly predicted images from all the four test classes to the total number of test images in all the four classes. It can be determined from the confusion matrix diagram of a model as the ratio of the summation of the diagonal elements to the total 320 test images.

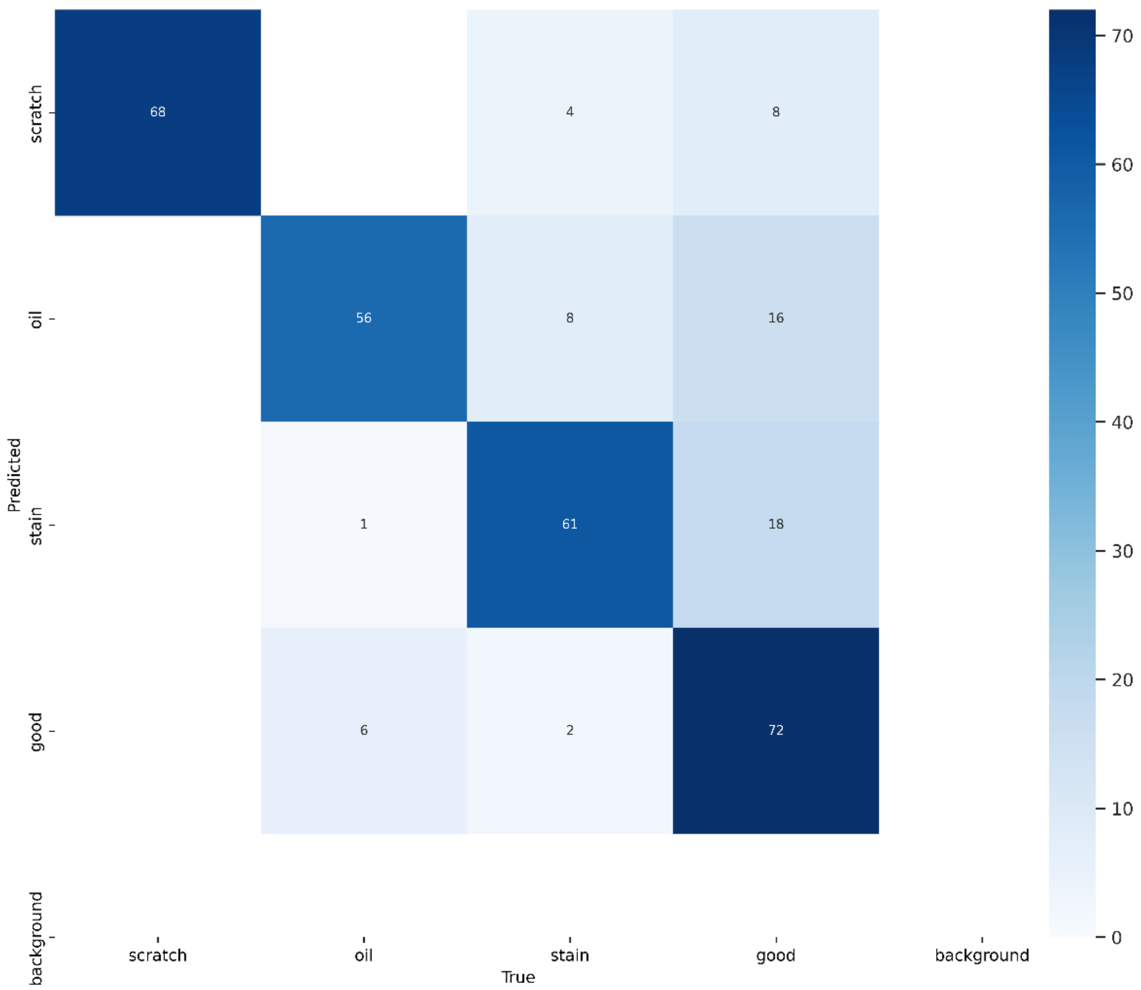

Figure 4.9: Confusion Matrix of VGG16 from Scratch Model

As can be seen from Figure 4.7, the highest validation accuracy of the trained VGG16 model is 80.3% and the accuracy saturates around 80%. This shows the selection of 50 epoch for training this model is right and increasing the number of epoch will not result in improvement of the performance of the model. The minimum validation loss is about 0.7 and it saturates at this value starting from the 20th epoch.

The confusion matrix of the VGG16 based model with the highest accuracy is shown in Figure 4.9. It is developed using the 320 test images with 80 images for each class. As it can be observed from the figure, 68 images are predicted correctly and classified under scratch from the 80 images in the scratch defect class. But, four images are predicted incorrectly and classified as if they are stain defects and the remaining eight images are predicted incorrectly as defect free in the good class.

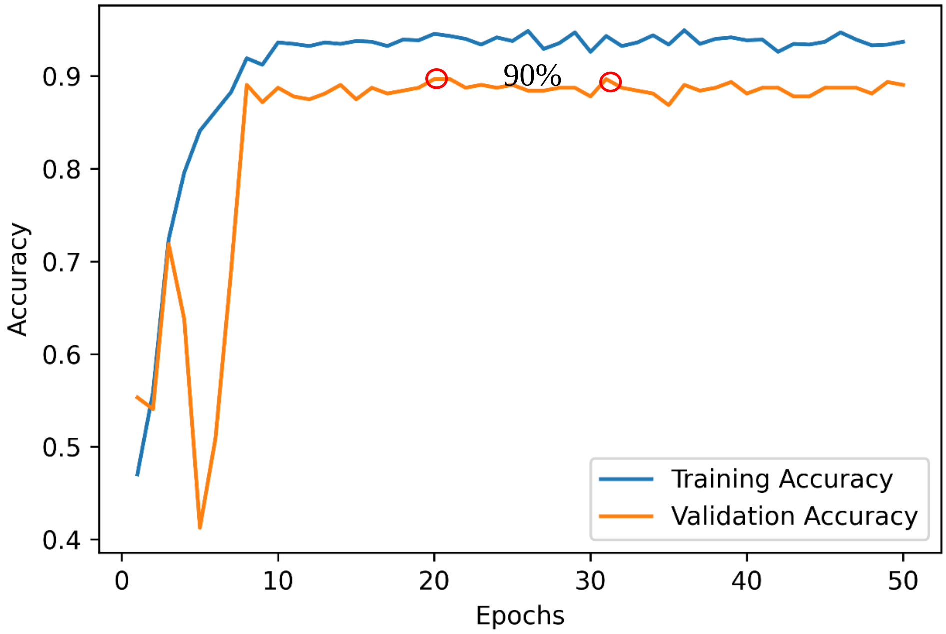

Figure 4.10: Training and Validation Accuracies of ResNet-34 from Scratch Model

From the oil class images 56 of them are predicted correctly and the remaining 24 images are predicted incorrectly. Eight images are predicted to be stain defects while the rest 16 images are predicted under good class. In the stain class, 61 of them are predicted correctly and 18 images are classified as stain while an image is predicted to be oil defect.

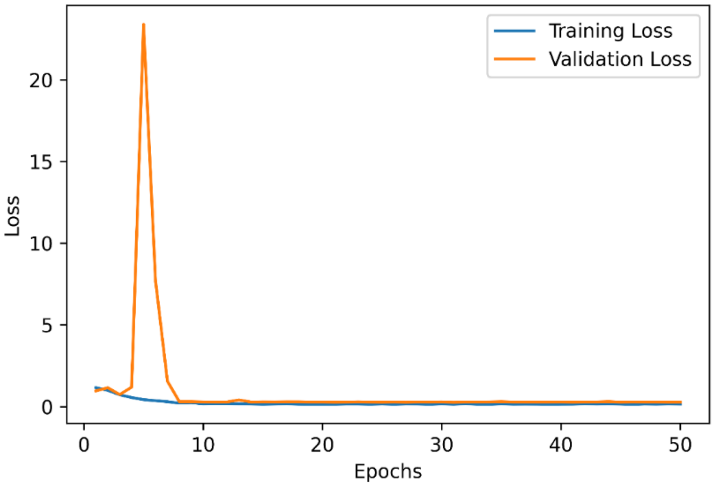

Figure 4.11: Training and Validation Losses of ResNet-34 from Scratch Model

From the good class images 72 of them are predicted correctly by this trained model and the remaining six and two images are predicted incorrectly in the oil and stain classes respectively.

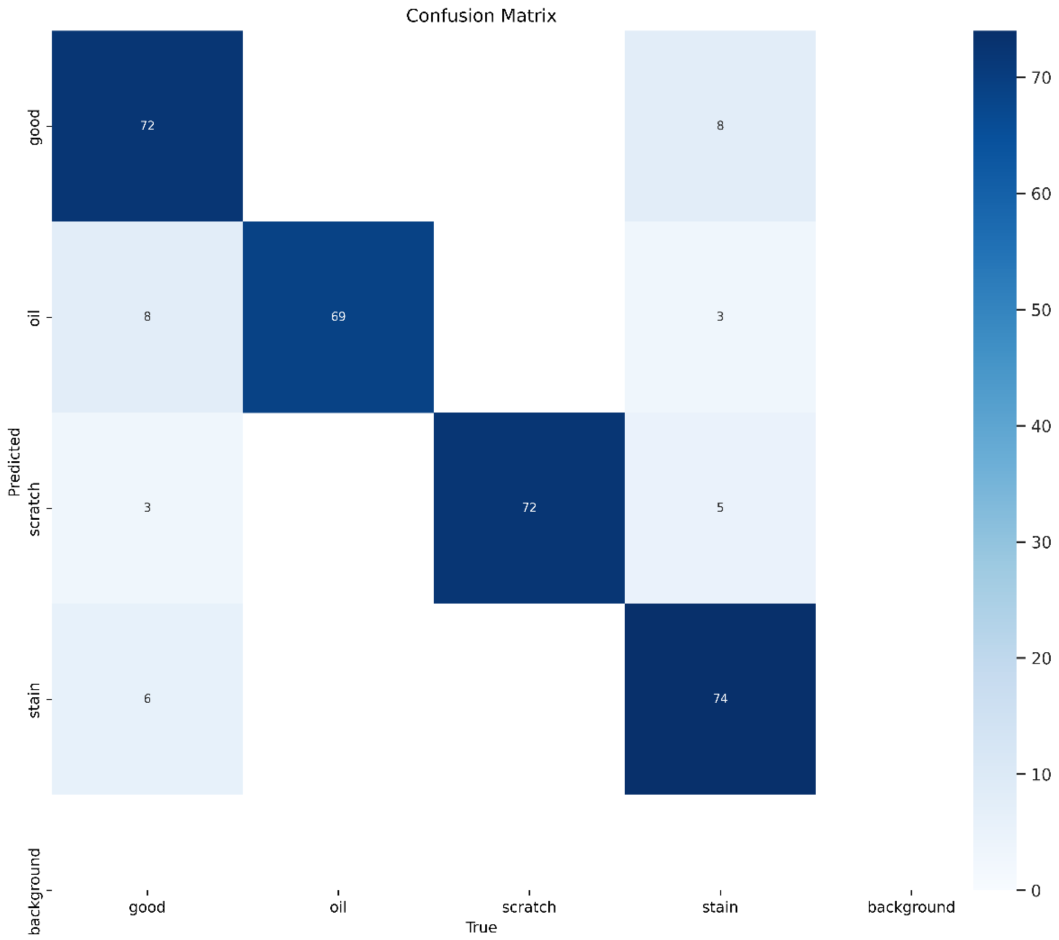

Figure 4.12: Confusion Matrix of ResNet-34 from Scratch Model

ResNet-34 Training and Testing Results

By using the same training and testing dataset with that of VGG16 model, ResNet-34 model is trained from scratch to build a new model. The training and validation accuracies and losses during the training process are shown in Figure 4.10 and Figure 4.11 respectively. The highest validation accuracy achieved is 90% and it saturates around this value starting from the eighth epoch.

About the third to the eighth epoch during training this model, there was some form of spike due to prediction error and inaccuracy. But after the eighth epoch, the model adjusts its parameters and the validation loss reduces to smaller values while the accuracy increases. This can easily be observed from Figure 4.10 and Figure 4.11.

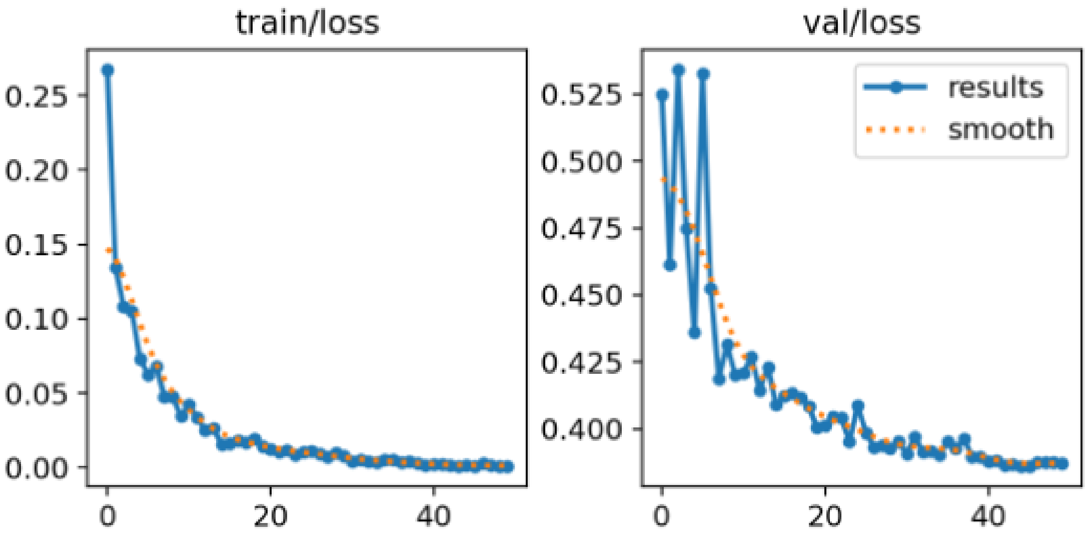

Figure 4.13: Training and Validation Losses of YOLOv8 from Scratch Model

The confusion matrix of this ResNet-34 based model with the highest accuracy is shown in Figure 4.12. The correct predictions for the good, oil, scratch and stain classes are 72, 69, 72 and 74 images respectively. Similar explanation with that of Figure 4.9 can be done to understand this confusion matrix diagram. For example, eight defect free good images are predicted by the model to be defective with stain defect.

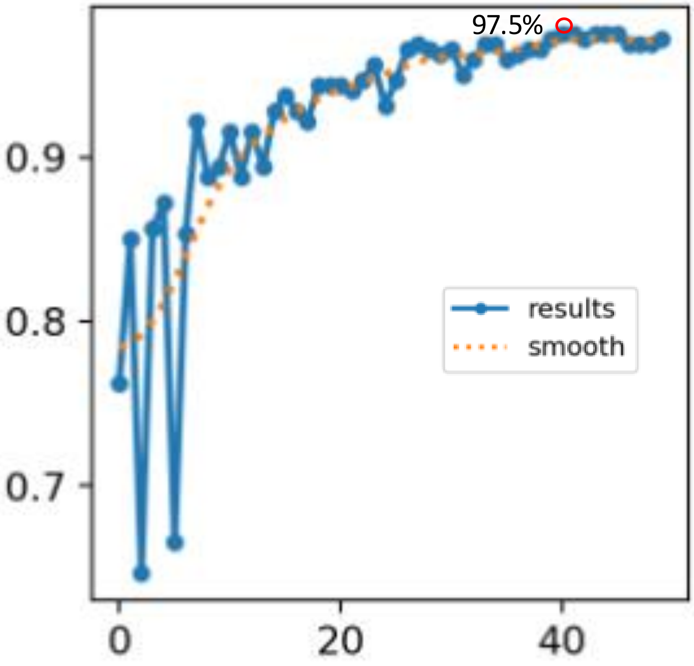

Figure 4.14: Validation Accuracy of YOLOv8 from Scratch Model

YOLOv8 Training and Testing Results

Using the same dataset used for the previous VGG16 and RseNet-34 based models, YOLOv8 model is trained from scratch. The training and validation losses and the validation accuracy are shown in Figure 4.13 and Figure 4.14 respectively.

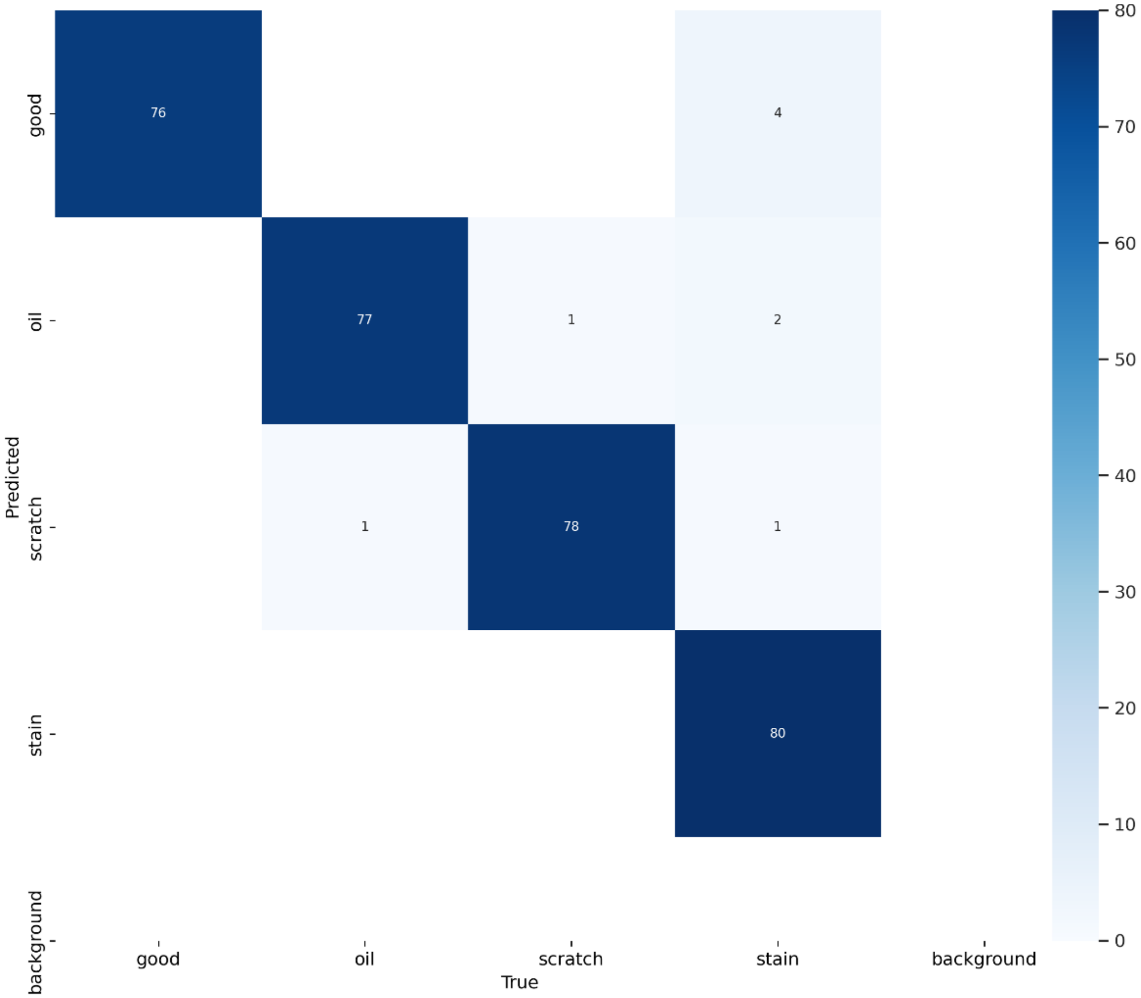

Figure 4.15: Confusion Matrix of YOLOv8 from Scratch Model at the Last Epoch

Because YOLOv8 is an optimized and recent model, it gives two output models after training. The first output is the model with the best accuracy and the second one is the model generated at the last epoch. The validation confusion matrix generated after training is for the model at the last epoch. This means, the best model will have a confusion matrix that shows better performance. The validation accuracy of the best model is found to be 97.5% as shown in Figure 4.14.

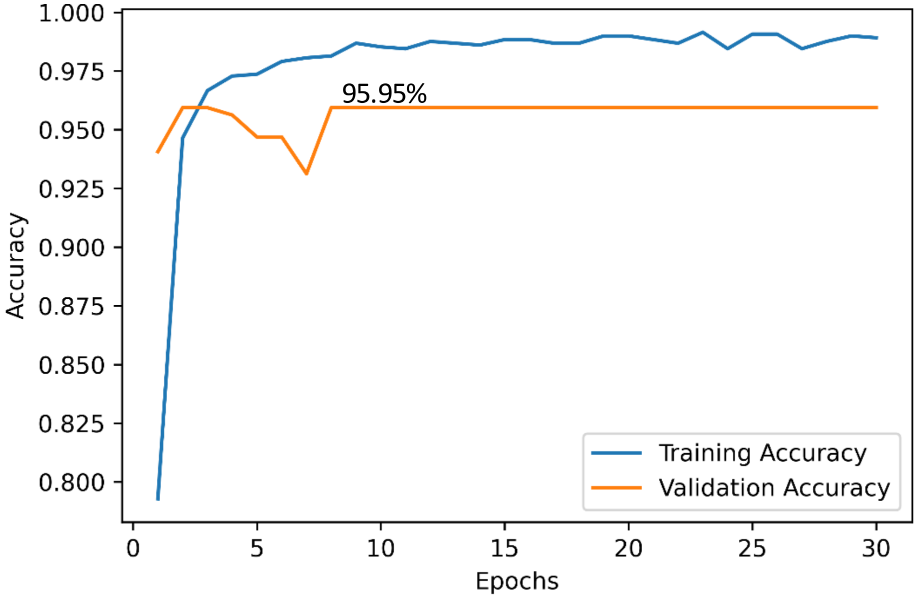

Figure 4.16: Training and Validation Accuracies of ResNet-34 from Pre-trained Model

As the values of the diagonal elements of the matrix increase or when the colors of these elements are darker, it means the performance of the model is better. The confusion matrix of the model at the last epoch is shown in Figure 4.15. The same explanation with that of Figure 4.9 can be done to understand this figure. As it can be observed from this figure, no defective image is classified as defect free image in the good class.

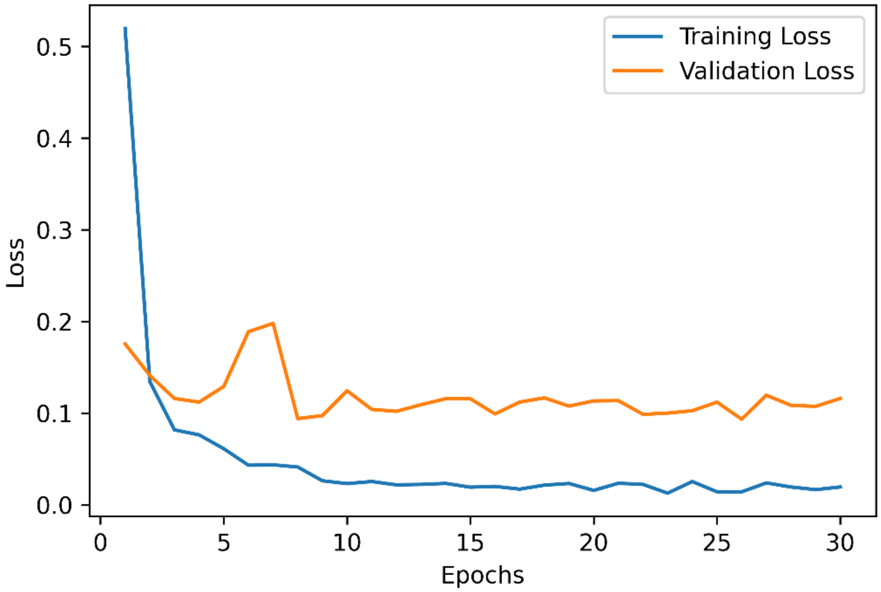

Figure 4.17: Training and Validation Losses of ResNet-34 from Pre-trained Model

Pre-trained ResNet-34 Training and Testing Results

To check whether the performance of the ResNet-34 and YOLOv8 based models can be improved, transfer learning is used by fine tuning pre-trained models. Both the ResNet-34 and YOLOv8 models are pertained by large image datasets and found from PyTorch library and Ultralitics respectively. As a result, the number of epoch used to train (fine tune) these models is selected to be 30.

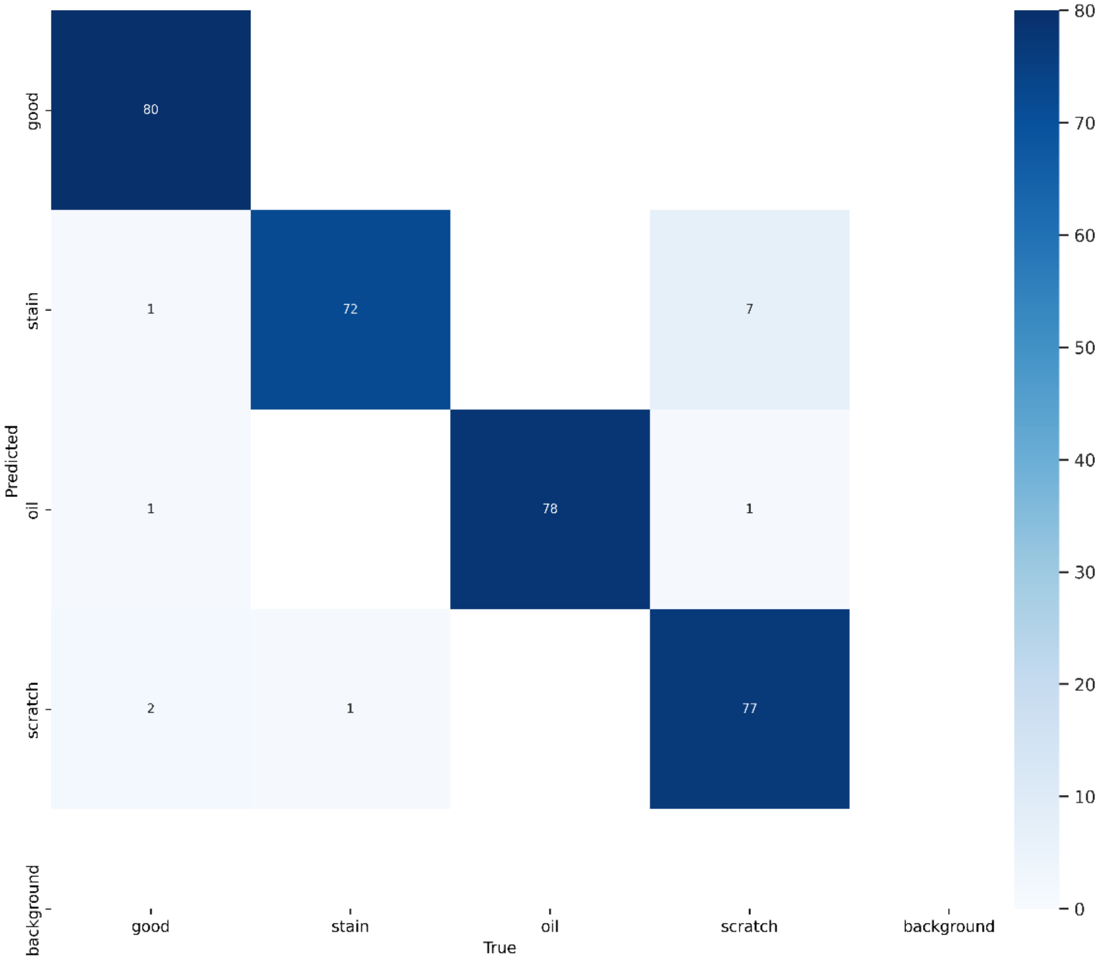

Figure 4.18: Confusion Matrix of ResNet-34 from Pre-trained Model

A pre-trained ResNet-34 model from PyTorch library is imported and trained again using the first dataset used to train the previous models. The fine tuning results of the pre-trained ResNet-34 model using this dataset is presented here as follows. The training and validation accuracies and losses are shown in Figure 4.16 and Figure 4.17 respectively.

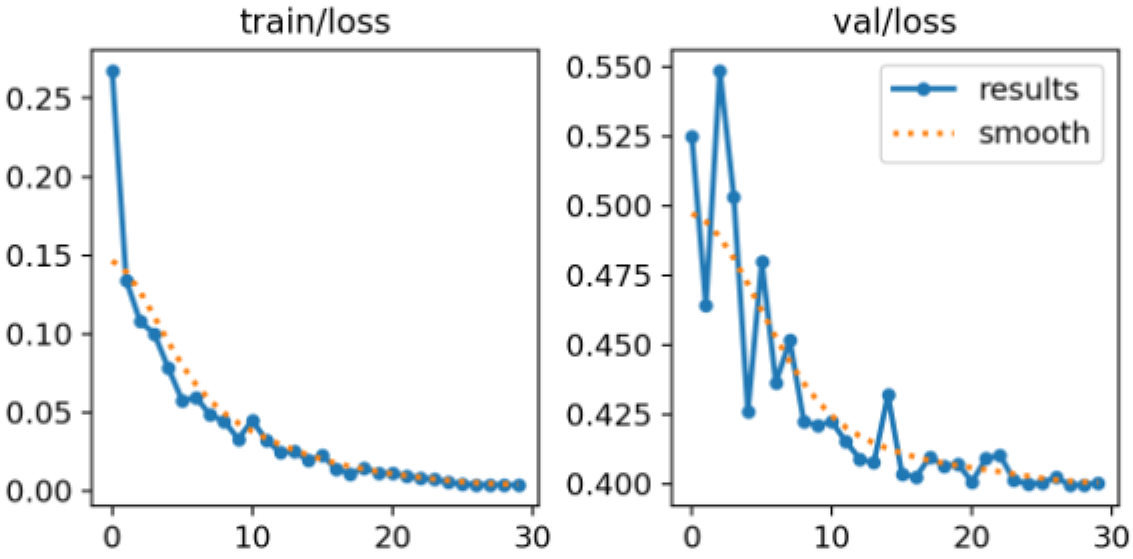

Figure 4.19: Training and Validation Losses of YOLOv8 from Pre-trained Model

As shown in Figure 4.16, the highest validation accuracy is 95.95% and the training process saturates at this value starting from about the eighth epoch. The confusion matrix of the trained (fine-tuned) model with 95.95% accuracy is shown in Figure 4.18. As it can be observed from the figure, this model has nice performance with no defective image classified as defect free in the good class. In addition to this the colors of the diagonal elements of the matrix are dark blue.

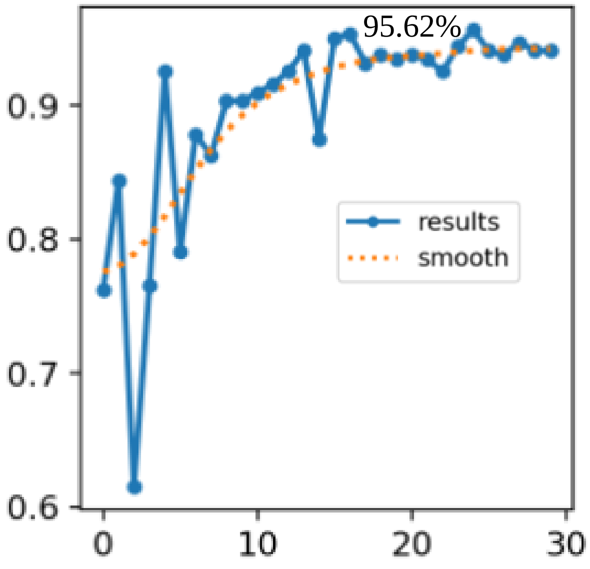

Figure 4.20: Validation Accuracy of YOLOv8 from Pre-trained Model

Pre-trained YOLOv8 Training and Testing Results

A pre-trained YOLOv8 model is imported from Ultralitics. Like that of the pre-trained ResNet-34 model, this pre-trained YOLOv8 model is trained using the first dataset. The training loss and the validation loss graphs during the training process is shown in Figure 4.19. The validation accuracy is also shown in Figure 4.20.

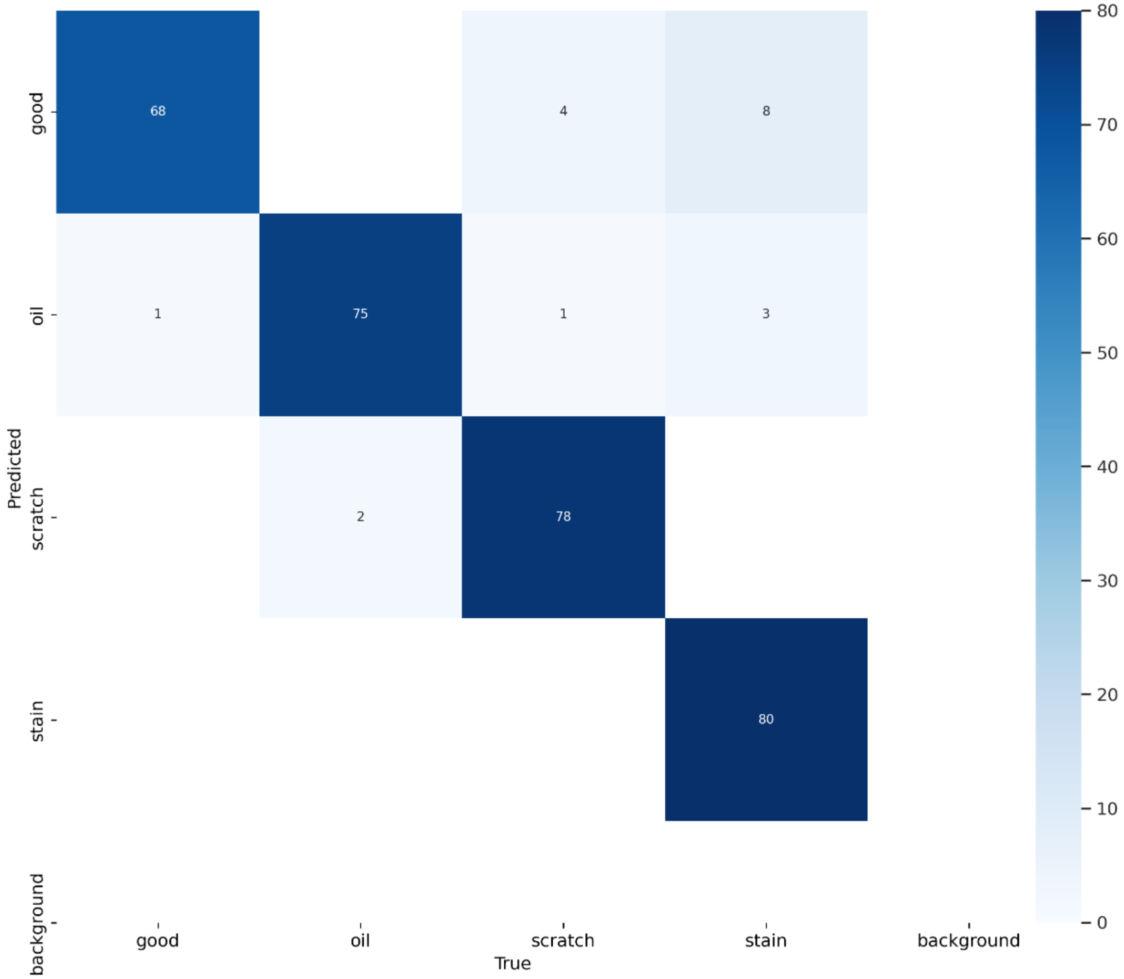

Figure 4.21: Confusion Matrix of YOLOv8 from Pre-trained Model at the Last Epoch

The highest validation accuracy achieved is 95.62%. The confusion matrix of the model at the last epoch of the training is shown in Figure 4.21. The model with the highest accuracy has better performance than that of the model generated at the last epoch.

The sixth experiment is done by using the second dataset to build a model through transfer learning. The fine-tuning is based on the pre-trained YOLOv8 model. This experiment is done for only demonstration purpose. It is because the size of the dataset used to train the pre-trained YOLOv8 model is too small for the trained model to have generalization capability. The pre-trained model is selected due to the small size of the dataset. The number of epoch used for this experiment is 50.

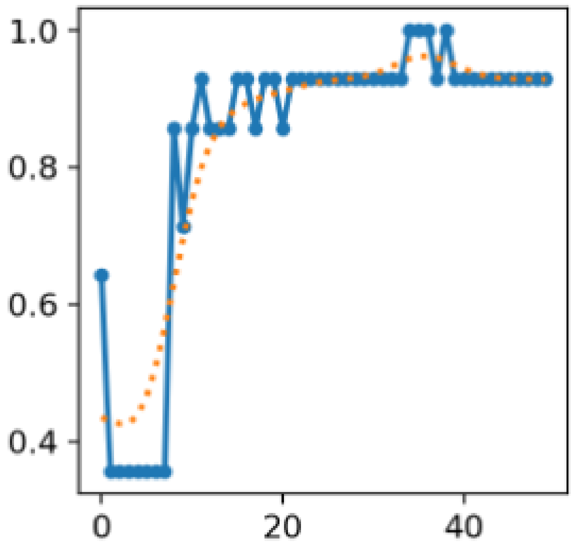

Figure 4.22: Validation Accuracy of YOLOv8 from Pre-trained Model Using the Second Dataset

As it can be seen from Figure 4.22, the highest validation accuracy is 100%. But, it may not be effective in predicting the defectiveness of new image. It has a threat of error for new unseen image.

Comparison of the Five Trained Models

The comparison of the five trained models based on the results on their confusion matrix is summarized in Table 4.2. For the YOLOv8 based models, the results are for the model at the last epoch. It means, the last epoch model accuracy is not the highest and the model with the highest accuracy will have a better performance.

In the table, the total detection means the total number of images predicted to be in a given class. The actual number is always 80 because the number of images in each class for testing are 80. When an image in one of the surface defect classes is predicted to be good, it is said to be miss detected. Therefore, miss detection means the total number of these miss detected images from a defect class by a model. Whereas, false classification is the total number of images when they are classified under incorrect defect class. Correct detection is the total number of images classified by the model correctly under their class. When this value is normalized by total detection in a given class, it will be correct detection rate.

Table 4.2: Test Results of Each Kind of Class

Metrics		**Actual Number**	**Total Detection**	**Correct Detection (Number)**	**Miss Detection (Number)**	**False Classification (Number)**	**Correct Detection (Rate)**
Good	VGG16_Scratch	80	114	72	0	8	63.2%
	ResNet34_Scratch	80	89	72	0	8	80.9%
	YOLOv8_Scratch	80	76	76	0	4	100%
	ResNet34_Pretrained	80	84	80	0	0	95.2%
	YOLOv8_Pretrained	80	69	68	0	12	98.6%
Oil	VGG16_Scratch	80	63	56	16	24	88.9%
	ResNet34_Scratch	80	69	69	8	11	100%
	YOLOv8_Scratch	80	78	77	0	3	98.7%
	ResNet34_Pretrained	80	78	78	1	2	100%
	YOLOv8_Pretrained	80	77	75	1	5	97.4%
Scratch	VGG16_Scratch	80	68	68	8	12	100%
	ResNet34_Scratch	80	72	72	3	8	100%
	YOLOv8_Scratch	80	79	78	0	2	98.7%
	ResNet34_Pretrained	80	85	77	2	3	90.6%
	YOLOv8_Pretrained	80	83	78	0	2	94%
Stain	VGG16_Scratch	80	75	61	18	19	81.3%
	ResNet34_Scratch	80	90	74	6	6	82.2%
	YOLOv8_Scratch	80	87	80	0	0	92%
	ResNet34_Pretrained	80	73	72	1	8	98.6%
	YOLOv8_Pretrained	80	91	80	0	0	87.9%

As can be observed from the table, the performance of the trained VGG16 model is the poorest. The performances of the trained models from pre-trained ResNet-34 and YOLOv8 from scratch are comparable with highest performances. The miss detection of the model developed from YOLOv8 from scratch is zero for all the three defect classes.

The comparison of these five trained models based on the model validation accuracy and the total number of trainable parameters is presented in Table 4.3. A trainable parameter is the parameter of an ML model that is updated after every iteration during a training process. The number of the trainable parameters directly affects the training and run time of the model as well as the computational resource required to train the model.

YOLOv8 has the lowest number of trainable parameters. As a result, the two YOLOv8 based models are the fastest models. In this table, the YOLOv8 based model that is trained from scratch is with the highest validation accuracy which is 97.5%. It is also the fastest model. For predicting the class of a single phone screen image, this model takes only 25 milliseconds.

Table 4.3: Comparison of the Five Models Based on Accuracy and No. of Trainable Parameters

Model	Accuracy	Number of Trainable Parameters
ResNet34_Scratch	90%	More than 21,000,000
ResNet34_Pretrained	95.95%	More than 21,000,000
VGG16_Scratch	80.80%	More than 133,000,000
YOLOv8_Scratch	97.50%	More than 1,400,000
YOLOv8_Pretrained	95.62%	More than 1,400,000

4.4 Framework for Developing ML Inspection Models

1. **Problem definition**: Clearly define the inspection problem that is required to be solved. What are the specific defects that are to be detected? What are the characteristics and natures of the defects that are to be modeled? More discussion of this stage is shown in section 4.1 above.
2. **Data collection**: Collect a dataset of images or sounds that contain the defects that are to be detected. The dataset should be large enough to train a robust ML model. Images of good and defective components or sound recordings of good and defective sound devices should be collected. This may take longer time because the availability of defective components and devices in a given production period say a month may be very small. Therefore, the focus should be in capturing the images of defective components in the image based inspection problems and in recording the sounds of defective sound devices in the sound based inspection problems. For image based inspection problems, at least five hundred images are required for every defect type. In order to develop robust ML model, it will be good to have thousands of images for every defect type. For the sound based inspection problems also, it is good to have thousands of different sound recordings for every defect type. But, in some sound inspection problems hundreds of sound recordings for each of the defect types may be enough. The size

and quality of the dataset affect the performance of the ML model. A larger dataset generally leads to a more accurate model, but it also require more time and resources to collect and prepare. For the images taken and collected as well as for the sound recordings, most of the labeling of the defect type for every data sample should be done at this stage.

3. **Dataset preparation (Preprocessing)**: Prepare the dataset for training the ML model. The purpose of preprocessing is to convert the data to a format that can be used by the model. This may involve tasks such as cleaning the data, removing outliers, image augmentation and normalization, removing noise and normalizing the audio levels of sound recordings, and feature extraction.
4. **Model selection**: Select an ML algorithm that is well-suited for the task of defect detection. There are many different ML algorithms available, so it is required to experiment to find the one that works best for the prepared dataset as demonstrated in section 4.3.
5. **Model training**: Train the ML model on the prepared dataset. This may take some time, depending on the size of the dataset and the complexity of the ML algorithm. This is also demonstrated in section 4.3.
6. **Model evaluation**: Evaluate the performance of the ML model on a held-out dataset. This is required to determine whether the model is accurate enough to be used in production. In the last sub section of section 4.3 this stage is thoroughly demonstrated.
7. **Model deployment**: Deploy the ML model in production. This may involve integrating the model into a manufacturing process or a quality control system.

4.5 Discussion of Results

This thesis work has four output results as discussed in the previous four sections. The first result is the output of the study of the existing inspection process of the company. From the study, it was found that the inspection methods adopted in the company are manual based. The main challenges in the inspection process are found to be inspection inaccuracies and slow inspection speed. These are shown by two sample cases. A one year inspection report showed that from the total defects found at the final inspection, more than 31% of the defects were missed from receiving inspection. The second sample case is about inspection speed. In order to inspect the surface defect of a cell phone screen, it takes a minimum of 3 seconds for a well-trained inspector.

The second result is the identification of the inspection problems that are best solved by ML based solutions. There are five classes of inspection problems in the inspection process as discussed in section 4.2. Most of them are found to be image based inspection problems. From literature review, the image based and the sound based inspection problems are found to be best solved by ML driven solutions. The best ML models for the image based problems are CNN class algorithms whereas for the sound based problems CNN and RNN class algorithms are best choices. The detailed model development procedures are also the outputs of the thesis work.

The third result is the phone screen surface defect detection ML model. As discussed in section 4.3, five ML models are built by training the combined dataset from MSD dataset and images taken from the company. The performances of the trained models are then compared. It is found that the adoption of ML driven solutions for the inspection problems is promising. The accuracy of the best YOLOv8 from scratch model is found to be 97.5% and the model can inspect a phone screen within 25 milliseconds. These results have shown that the proposed ML models have the potential to curb the challenges encountered in the inspection process.

The pre-trained YOLOv8 model that is trained using the dataset prepared from images taken from the company, has also shown good results. This trained model has an accuracy of 100%, but it is prone to error threat due to the small size of the dataset. If a large enough dataset is prepared and used, the error threat can be avoided.

The fourth result is the framework presented in the previous section. Using this framework the case company can develop ML based models for the image based and sound based inspection problems. Once the developed model is evaluated and its performance is good, the next step is to deploy it to production. This may be integrating in a conveyor, robotic or some other automation system.

Most of the inspection problems at the receiving inspection can be implemented and integrated in simple conveyor systems. In addition to these, the surface defect detection for all the cell phone components can be implemented and integrated in simple conveyor system. In the case company, there is a conveyor system which is not functional due to the slow speed in the receiving inspection. By developing ML models for these inspection problems, the models can be deployed and integrated with this existing conveyor system.

The visual based functional tests in the in-process inspection and the sound based inspection problems may require advanced robotic systems for ML model deployment. This is because, these inspection problems require interaction of the inspection system through touch screens and key pads.

Chapter 5 Conclusion and Recommendation

5.1 Conclusion

In this thesis work, the quality inspection process in a cell phone assembling company in Ethiopia is studied to make use of ML driven inspection solutions. After studying the inspection process in detail, the inspection problems that can be solved by ML driven approaches are identified. These are image based and sound based inspection problems.

Based on literature study, it is concluded that the image based and sound based inspection problems are best suited for ML based approaches. For the image based inspection problems CNN class algorithms are proposed while for the sound based inspection problems CNN and RNN class algorithms are suggested.

To demonstrate the potential of the ML driven approaches, five ML models are trained and developed for phone screen defect detection and classification. The performances of the developed models are compared and the YOLOv8 based model that is trained from scratch is found to be with the highest performance. The speed and accuracy performances of these ML models are found to be promising. From these results, it is concluded that ML driven inspection approaches have high potential in resolving the inspection challenges of the company. But, better performance is achieved, if a large dataset of different types of defective and defect free components is prepared and used.

5.2 Recommendation

Most of the quality inspection problems in the case company are visual based inspections. For the sensitive components inspection like that of PCBA, inspecting subtle defects visually through human eye is challenging. Such inspection problems are recommended to be solved by ML based solutions. But, in order to build high performance ML inspection model, it is required to have a large volume dataset. Therefore, the company is highly recommended to take the necessary documentation of the defective components before disassembling and scraping.

By collecting actual data from the company for the sensitive components and labeling it manually, a large volume of realistic dataset can be prepared. Using such dataset, high performance ML models can be developed. This work is recommended to other researchers as a future work.

Reference

Babic, M., Farahani, M. A., & Wuest, T. (2021). Image Based Quality Inspection in Smart Manufacturing Systems: A Literature Review. *Procedia CIRP*, *103*, 262–267. https://doi.org/10.1016/j.procir.2021.10.042

Broday, E. E. (2022). The evolution of quality: from inspection to quality 4.0. *International Journal of Quality and Service Sciences*, *14*(3), 368–382. https://doi.org/10.1108/IJQSS-09-2021-0121

Du, S. (2021). Research on Image Recognition Algorithm Based on Computer Deep Learning. *Journal of Physics: Conference Series*, *1992*(3). https://doi.org/10.1088/1742-6596/1992/3/032065

Guliti, M. (2018). Article on Assessment of Performance Appraisal Practices in Ethiopia. *International Journal of Trend in Research and Development*, *5*(2), 638–644.

Hussain, M., Bird, J. J., & Faria, D. R. (2019). A study on CNN transfer learning for image classification. *Advances in Intelligent Systems and Computing*, *840*(October), 191–202. https://doi.org/10.1007/978-3-319-97982-3_16

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

Lee, I., & Shin, Y. J. (2020). Machine learning for enterprises: Applications, algorithm selection, and challenges. *Business Horizons*, *63*(2), 157–170. https://doi.org/10.1016/j.bushor.2019.10.005

Lei, J., Gao, X., Feng, Z., Qiu, H., & Song, M. (2018). Scale insensitive and focus driven mobile screen defect detection in industry. *Neurocomputing*, *294*, 72–81. https://doi.org/10.1016/j.neucom.2018.03.013

Li, C., Zhang, X., Huang, Y., Tang, C., & Fatikow, S. (2020). A novel algorithm for defect extraction and classification of mobile phone screen based on machine vision. *Computers and Industrial Engineering*, *146*(November 2019), 106530. https://doi.org/10.1016/j.cie.2020.106530

Li, L. (2020). Application of deep learning in image recognition. *Journal of Physics: Conference Series*, *1693*(1). https://doi.org/10.1088/1742-6596/1693/1/012128

Miles, B., & Surgenor, B. (2011). Machine Vision for Inspection: A Case Study. *Assembly Line - Theory and Practice*. https://doi.org/10.5772/21546

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

Rahman, M. A. A., Rahman, M. H., Mohamad, E., Rahman, A. A. A., Salleh, M. R., & Mo, J. P. T. (2019). Enhancing the automated quality inspection in manufacturing process through parameter optimization. *Universal Journal of Mechanical Engineering*, *7*(6), 1–5. https://doi.org/10.13189/ujme.2019.071501

Ray, S. (2019). A Quick Review of Machine Learning Algorithms. *Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019*, 35–39. https://doi.org/10.1109/COMITCon.2019.8862451

Reichenstein, T., Raffin, T., Sand, C., & Franke, J. (2022). Implementation of Machine Vision based Quality Inspection in Production: An Approach for the Accelerated Execution of Case Studies. *Procedia CIRP*, *112*, 596–601. https://doi.org/10.1016/j.procir.2022.09.058

Sachan, V., Deb, P. S., Sharma, S., Thakur, H., & Dubey, G. K. (2020). Product quality inspector. *Proceedings of the 2020 9th International Conference on System Modeling and Advancement in Research Trends, SMART 2020*, 212–215. https://doi.org/10.1109/SMART50582.2020.9336802

Schmitt, J., Bönig, J., Borggräfe, T., Beitinger, G., & Deuse, J. (2020). Predictive model-based quality inspection using Machine Learning and Edge Cloud Computing. *Advanced Engineering Informatics*, *45*(May 2019), 101101. https://doi.org/10.1016/j.aei.2020.101101

Selvamuthu, D., & Das, D. (2018). Introduction to statistical methods, design of experiments and statistical quality control. In *Introduction to Statistical Methods, Design of Experiments and Statistical Quality Control*. https://doi.org/10.1007/978-981-13-1736-1

Sun, X., Wang, B., Wang, Z., Li, H., Li, H., & Fu, K. (2021). Research Progress on Few-Shot Learning for Remote Sensing Image Interpretation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *14*, 2387–2402. https://doi.org/10.1109/JSTARS.2021.3052869

Sung, F., Yang, Y., & Zhang, L. (2018). Relation Network for Few-Shot Learning. *Cvpr*, 1199–1208.

Wang, J., Fu, P., & Gao, R. X. (2019). Machine vision intelligence for product defect inspection based on deep learning and Hough transform. *Journal of Manufacturing Systems*, *51*(December 2018), 52–60. https://doi.org/10.1016/j.jmsy.2019.03.002

Weiss, K., Khoshgoftaar, T. M., & Wang, D. D. (2016). A survey of transfer learning. In *Journal of Big Data* (Vol. 3, Issue 1). Springer International Publishing. https://doi.org/10.1186/s40537-016-0043-6

Xiao, Q., Liang, S., & Huang, J. (2022). Quality inspection system for metal parts based on machine vision. *Second International Conference on Advanced Algorithms and Signal Image Processing (AASIP 2022), November 2022*, 29. https://doi.org/10.1117/12.2659356

Yang, X., Zhang, Y., Lv, W., & Wang, D. (2021). Image recognition of wind turbine blade damage based on a deep learning model with transfer learning and an ensemble learning classifier. *Renewable Energy*, *163*, 386–397. https://doi.org/10.1016/j.renene.2020.08.125

Yavari, N. (2020). *Few-shot Learning with Deep Neural Networks for Visual Quality Control: Evaluations on a Production Line*.

Zakaria, S. S., Amir, A., Yaakob, N., & Nazemi, S. (2020). Automated Detection of Printed Circuit Boards (PCB) Defects by Using Machine Learning in Electronic Manufacturing: Current Approaches. *IOP Conference Series: Materials Science and Engineering*, *767*(1).

https://doi.org/10.1088/1757-899X/767/1/012064

Zonnenshain, A., & Kenett, R. S. (2020). Quality 4.0—the challenging future of quality engineering. *Quality Engineering*, *32*(4), 614–626. https://doi.org/10.1080/08982112.2019.1706744