OPTIMIZATION OF SCREW PRODUCTION USING DEEP CONVOLUTIONAL NEURAL NETWORK (DCNN)

BY

NDUKWE CHIDIOGO DANIEL (B.Eng)

20174079908

A THESIS SUBMITTED TO THE POST GRADUATE SCHOOL

FEDERAL UNIVERSITY OF TECHNOLOGY, OWERRI

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE MASTER DEGREE (M.Eng) IN INDUSTRIAL PRODUCTION ENGINEERING, DEPARTMENT OF MECHANICAL ENGINEERING.

MARCH, 2024.

Certification

This is to certify that this work "Optimization of Screw Production using Deep Convolutional Neural Network" was carried out by I, Ndukwe Chidiogo Daniel (20174079908) in partial fulfillment for the award of the degree of Master of Engineering in Industrial Production Engineering, Department of Mechanical Engineering of the Federal University of Technology, Owerri.

………………… 05/06/2024

Engr. Dr. O.O.Obiukwu — Date

Supervisor

………………… 05/06/2024

Dr. D.O. Njoku — Date

Co-supervisor

………………… 05/06/24

Engr. Dr. G.O.Onuoha — Date

Head of Department

………………… 13/6/24

Engr. Prof. Remy Uche — Date

Dean, SEET

………………… …………………

Prof. B.O.Eseonu — Date

Dean, School of Postgraduate

………………… …………………

Engr. Prof. H.U. Nwosu — Date

External Supervisor

Dedication

This work is dedicated to God Almighty, whom without His infinite goodness and mercy, none of these would have possible.

Acknowledgments

My profound gratitude goes to God Almighty who has made it a possible for me to conclude this work, it was filled with ups and downs, but He led me through. I am ever grateful to my supervisor Engr. Dr. O. Obiukwu, for the tremendous effort and endless disturbances he had to put up with to ensure this work was a success. It was through his tutelage and guidance that I was able to complete this work. To my Co-supervisor, Dr. D.O. Njoku, I truly appreciate all your encouragement, support and guidance, even when it was not convenient for you, thank you sir. To all my lecturers that drilled me through this course, Engr. Prof. G.O.Osueke, you ensured I understood what I was doing, thank you for all the lectures you made us present; Engr. Prof. A.C. Uzorh, I was privileged to sit under your tutelage, your wealth of knowledge and wisdom is profound, thank you very much sir; Engr. Prof. Remy Uche, thank you sir for always making me demand more of myself, thank you very much sir, your effort and encouraging words were our pillar in difficult times; Engr. Prof.Okoronkwo, you always made yourself available and guided me properly during my course work, I sincerely appreciate you sir; Engr. Dr. G.O.Onuoha our able, friendly and supportive H.O.D, thank you very much sir.

To my Dean, S.E.E.T., Engr. Prof. Remy Uche and the Dean of post graduate school, Prof. B.O.Eseonu. I truly appreciate this opportunity you have given to me to further develop and master in my field.

To my course mates Engr. (Mrs) R. Paul-Okore, Engr. J. Chukwu, MrChimaDimson, MrFinian, Mr Charles, MrsNkechiEzeaku, Mr Great, MrAnyawuMicheal, Mr Mac-Anthony, I appreciate all your support, it was vital in the long run.

It is paramount that I appreciate Sir and Lady I.D. Ndukwe, my parentss for their numerous supports and prayers all through my life till date.Finally, I appreciate the efforts of all who have been helpful through other means such as ideas, finance and words of encouragement.

TABLE OF CONTENTS **PAGE**

LIST OF FIGURES PAGE

LIST OF TABLES PAGE

ACRONYMS

ANN = ARTIFICIAL NEURAL NETWORK

NN = NEURAL NETWORK

CPU = CENTRAL PROCESSING UNIT

PNN = PROBABILISTIC NEURAL NETWORK

GMM = GENERALIZED METHODS OF MOMENTS

BPN = BACK-PROPAGATION NEURAL NETWORK

CBR = CASE-BASED REASONING

CNN = CONVOLUTIONAL NEURAL NETWORK

GPU = GRAPHICS PROCESSING UNIT

FC = FULLY CONNECTED

IoT = INTERNET OF THINGS

AOI = AUTOMATED OPTICAL INSPECTION

GAN = GENERATIVE ADVERSARIAL NETWORK

DL = DEEP LEARNING

ML = MACHINE LEARNING

AAE = ADVERSARIAL AUTOENCODER

CAE = CONVOLUTIONAL AUTOENCODER

AI = ARTIFICIAL INTELLIGENCE

MLP = MULTI-LAYER PERCEPTRON

DNN = DEEP NEURAL NETWORK

SGD = STOCHASTIC GRADIENT DESCENT

RBM = RESTRICTED BOLTSMANN MACHINE

AE = AUTOENCODER

CCD = CEMERA SENSING DEVICE

NCC = NORMALIZED CROSS-CORRELATION

IoU = INTERSECTION OVER UNION

DC = DICE COEFFICIENT

FPS = FRAMES PER SECOND

ReLU= RECTIFIED LINEAR UNIT

ABSTRACT

This research proposed a deep convolutional neural network (DCNN) based technique for the detection of micro defects on metal screw surfaces. Defects considered include surface damage, surface dirt, and stripped screws. Images of metal screws with different types of defects were collected using industrial cameras, which were then employed to train the designed deep CNN. To enable efficient detection, I first located screw surfaces in the pictures captured by the cameras, so that the images of screw surfaces could be extracted, which were then inputtedinto the CNN-based defect detector. Experiment results showed that the proposed technique could achieve a detection accuracy of 97%; the average detection time per picture is 1.2 seconds. Comparisons with traditional machine vision techniques, e.g., template matching-based techniques, demonstrate the superiority of the proposed deep CNN-based one. Furthermore, it could be seen that the accuracy of the proposed DCNN was much higher than the traditional LeNet-5 at the beginning of the network training and the accuracy of the training was to 100% with 550 iterations and about 100% accuracy was achieved with 800 iterations.

Keywords: Screw image; Deep Convolutional Neural Network; Micro-defect Detection; Internet of Things.

CHAPTER ONE

INTRODUCTION

1.1 Background to the Study

In the industrial production processes of the fastener industry, several defects can occur and cause high manufacturing costs as well as a high amount of rejects (Song et al., 2018). The quality inspection of screws is a critical process due to complex structure and variety of defects (Dong et al., 2018). Screws are indispensable elements of many mechanical components, and it is crucial that those elements do not negatively affect the product. Therefore, it is necessary to inspect all manufactured screws (Ukida, 2007). Screw parts like threads are one of the most important components of machinery. Threaded elements amounts to 15% of the mechanical parts in equipment and machines. In industrial production, the usage of screw threads is very high. Real-time and high-precision measurement, as well as the improvement of accessories in this field, is important (Gadelmawla, 2017).

One of the most popular quality management systems globally is ISO 9001, which aims to improve processes, access to foreign markets, and increase competitiveness. Another goal is an optimum quality level which equals zero defects. Many manufacturers follow the zero-defect strategy for their products. Hence, the total cost of quality can be decreased with quality improvement processes (Priede, 2012). Thus, it is necessary to perform quality control to detect defect screws in order to reduce manufacturing costs and to improve the quality and yield (Song et al, 2018). A high level of quality in products is necessary for manufacturers to be competitive globally and efficient (Rezaei-Malek et al., 2020). The detection of defects by manual work is costly, time-consuming as well as low efficiency. It can also lead to a high error rate, and not all produced screws can be inspected (Chen et al., 2018). Especially for a large number of screws that are produced in a short time, a high-speed inspection is necessary (Ukida, 2007).

For an automated classification and quality inspection of the produced goods, an Automated Visual Inspection (AVI) system is often used to replace the manual quality inspection (Prabuwono et al, 2019). The system recognizes the products lying on the conveyor belt and uses an algorithm for object detection and recognition (Arsovskia et al., 2019). Automatic defect recognition is a reliable method for quality control in the production processes (Dong et al., 2018). Thus, visual analysis for surface defect detection is a standard method (Wu et al, 2017). This important part of quality control is becoming more interesting for industrial manufacturing processes (Wang et al, 2018). Sensors and the Internet of Things cause an increase in data recorded in the manufacturing process. Machine learning approaches have proven to be an effective tool for the evaluation of different sensor data (Gaetner et al., 2021). Especially Deep Learning shows superior performance for the analysis of image data (Wang, 2018).

As industrial automation advances, Deep Learning can help simplify the process of quality control. Based onimages, defects in the manufactured objects can be detected, and defect products can be sorted out (Staar et al., 2019). Also, it contributes to more reliable and efficient defect detection in terms of the zero-defect strategy (Martinez et al., 2020). Therefore, good results were already achieved with Deep Learning approaches such as the work by Song et al. (2018), where a Convolutional Neural Network (CNN) approach was successfully used to detect micro defects of metal screw surfaces with images collected from an industrial camera.

However, no realistic production scenario was considered. Wu et al. (2019) concentrated on surface defect detection and created an adaptable CNN-based model. Nevertheless, the location of different defects was not considered like they can occur in the quality inspection (Bergmann et al., 2019). There are already CNN-based approaches for the defect detection of small parts for manufacturing, which have considered more realistic images. The work by Yang et al. (2019) considered the real-world manufacturing constraints and parameters for four different defect classes.

However, separate models were used for the different defects, which is not suitable for real production conditions in quality control. The various studies show that a CNN-based approach is an auspicious method for defect detection of screws in quality control. CNNs are able to identify defects based on automatically learned deep features and do not have to be designed by human engineers (LeCun et al., 2015). Therefore, this work investigated if a CNN approach could classify different defects in screws and defect-free screw images with realistic angles of lying screws and five defect types according to a real-world industrial inspection scenario. To achieve this, the Deep Learning method CNN with VGG16 (Visual Geometry Group) architecture is applied to test and train the screw data set (Bergmann et al., 2019). As several defects can occur in the manufacturing process, it is necessary to focus on defects that can happen in real-world manufacturing scenarios. The threads as an important part should therefore be considered as well as other common defects like scratches and dents on the screw head, top or neck (Gadelmawla, 2017).

In order to be able to detect realistic defects reliably, we chose a real-world scenario approach. Other approaches detect differences in the size of screws (Lehr et al., 2019) or classify different screw types (Yildiz and Worgotter, 2019), as well as identify several similar items like screws, washers, and nuts (Dong et al., 2018). Some focus on the detection of defects in different materials such as texture, metallic gasket, and screws but consider only one defect type (Wu et al., 2019) or detect defects of screw surfaces using only top view images (Song et al., 2018). I focused on recognizing five screw defects with side-view images like in a realistic production scenario. In this case, the screw front, head, neck, thread side, and thread top were considered, and defects like scratches and dents on the surface as well asdistorted and missing object parts could be recognized.

1.2 Problem Statement

Although the automated visual inspection (AVI) technique is broadly applied using a sorting machine to inspect for defective screws, the detection of the defective screws with a high degree of precision is still a challenging issue. At present, the texture analysis carried out by a computer vision algorithm of Fourier-based restoration is widely used to identify defective screw surfaces. The idea of Fourier transformation method transfers the thread image into a frequency domain. Then, the notch-rejected filter is used to eliminate the high-energy frequency of thread pattern and transform it back to the spatial domain, for the defective internal thread to be detected.

However, the limitation of the Fourier transformation method is that the thread pattern with different densities has a distinct frequency, leading to the tedious work of adjusting the parameters of the algorithm.

Owing to the mutual restraint of the complex algorithm parameters, the parameters variables are highly dependent on the production environment, such as; inhomogeneous illumination, low contrast, and blurry contour, resulting in the instability of detection results.

If different parameters are set, the results may be overkill (potential good units being killed) or under-kill (potential bad units escaping) of defective images. Moreover, the parameter setting of these complex algorithms requires well-trained professional operators to constantly adjust the parameters, which is time-consuming and tiresome task.

1.3 Objectives of the Study

The main objective of this work was the optimization of screw production using Deep Convolutional Neural Network (DCNN) in order to reduce manufacturing costs and to improve quality and yield. The specific objectives were:

i) To develop an automated classification method based on a CNN model with VGG16 architecture for a reliable classification of defective and non-defective screws.

ii) To ensure that DCNN-based approach could automatically detect faulty screws as part of camera-based in-line inspection during production using only a single image of the object.

iii) To validate and compare the designed and developed the ANN system with other existing models.

1.4 Justification of the Study

This justification highlights the importance of defect detection in manufacturing, the innovation of the deep CNN-based technique, and the substantial improvements it offers over traditional methods, which as follows:

i. **Significance of Defect Detection in Manufacturing**: Defect detection is vital for maintaining high-quality industrial production processes. Manufacturing defects lead to increased production costs and quality issues. Ensuring the quality of essential components like screws is critical to overall product reliability.

ii. **Need for Advanced Defect Detection Techniques**: Detection of various screw defects, including surface damage, dirt, and stripped screws, is a pivotal process. Current methods require enhanced accuracy and efficiency to address the complexity of defects in manufacturing.

iii. **Deep Convolutional Neural Network (CNN) Proposal**: This work introduced a novel approach using a deep CNN for precise detection of micro defects on metal screw surfaces. Deep CNNs are well-suited for image analysis, offering the potential for high accuracy in defect identification.

1.5 Scope of the Study

This work investigated if a CNN approach could classify different defects in screws and defect-free screw images with realistic angles of lying screws.Five defect types were evaluated according to real-world industrial inspection scenario. To achieve this, the Deep Learning method CNN with a VGG16 architecture was applied to test and train the screw data set.

CHAPTER TWO

LITERATURE REVIEW

2.1 Programming of Deep Convolutional Neural Network (DCNN)

A traditional convolutional neural network is made up of single or multiple blocks of convolution and pooling layers, followed by one or multiple fully connected (FC) layers and an output layer. The convolutional layer is the core building block of a CNN. This layer aims to learn feature representations of the input. The convolutional layer is composed of several learnable convolution kernels or filters which are used to compute different feature maps. Each unit of feature map is connected to a receptive field in the previous layer. The new feature map is produced by convolving the input with the kernels and applying elementwise non-linear activation function on the convolved result. The parameter sharing property of convolutional layer reduces the model complexity.

Pooling or sub-sampling layer takes a small region of the convolutional output as input and down-samples it to produce a single output. There are different sub-sampling techniques as example max pooling, min pooling, average pooling, etc. Pooling reduces the number of parameters to be computed as well as it makes the network translationinvariant (Louis, 2022). Last part of CNN is basically made up of one or more FC layers typically found in feed-forward neural network. The FC layer takes input from the final pooling or convolutional layer and generates final output of CNN. In case of image classification, a CNN can be viewed as a combination of two parts: feature extraction part and classification part. Both convolution and pooling layers perform feature extraction.

As an example of dog's image, different convolution layers from lower level to higher level detect various features such as two eyes, long ears, four legs, etc for further recognition. On top of these features, the FC layers are added as classifier, and a probability is assigned for the input image being a dog. Beside the layer design, the improvement of CNN depends on several different aspects such as activation function, normalization method, loss function, regularization, optimization and processing speed, etc.

After the success of AlexNet, CNN got huge popularity in three major fields namely; image classification, object detection and segmentation, and many advance models of CNN has been proposed in those areas in successive years. major application areas that apply CNN to achieve state-of-the-art performance includes image classification, object tracking, object detection, segmentation, human pose estimation, text detection, visual saliency detection, action recognition, scene labeling, visual question answering, speech and natural language processing, etc. Though current CNN models work very well for various applications, it is yet to know why and how it works essentially. So, more effort on investigating the fundamental principles of CNNs is required. The chapter will provide a better understanding of CNN as well as facilitates for future research activities and application developments in the field of CNN (Louis, 2022).

2.1.1 CNN Fundamentals

Understanding the various CNN components and their applications is critical to comprehending the advancements in CNN architecture. Figure 2.1 displays several CNN parts.

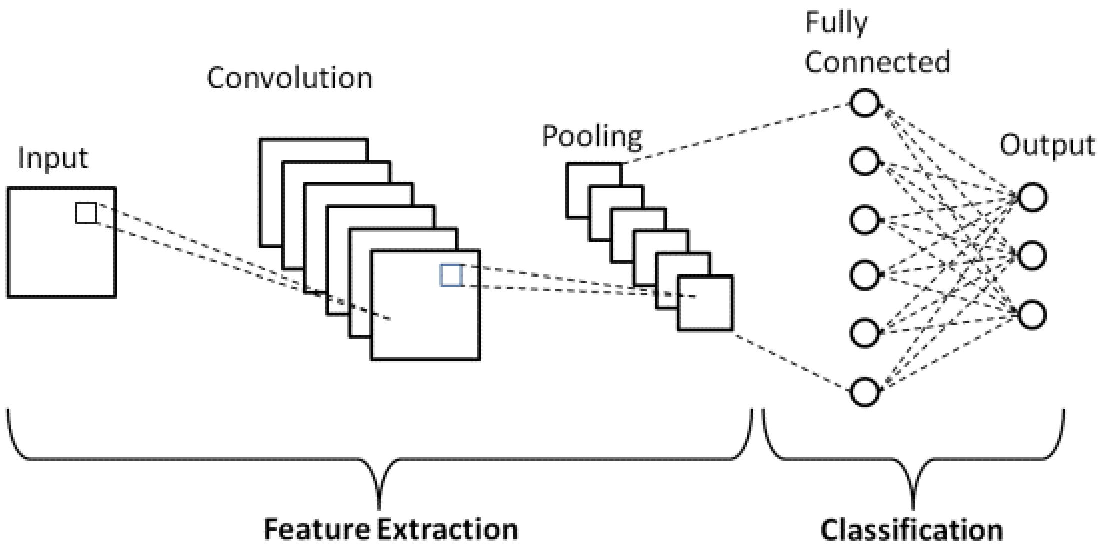


Figure 2.1: Binary Image classifier CNN using Tensor-Flow (Source: SaiBalaji, 2020.)

2.1.2 CNN Layers

A CNN is typically composed of four types of layers:

i.) Convolutional.

ii.) Pooling.

iii.) Function of Activation.

iv.) Fully Connected.

2.1.3 Convolutional Layer

The convolutional layer is a crucial part of CNN's overall structure. It is a set of filters or kernels applied to the data before it is used. Each kernel's width, height, and weight are used to extract characteristics from the input data. Weights in the kernels are first assigned at random but gradually become more informed by training data. (Koushik, 2016). A kernel is a set of discrete values or integers. For each number, the kernel weight is given as a reference. The initial kernel weight for a CNN is a set of integers picked at random. In addition, the weights are initialized in various ways. In turn, the kernel learns to extract meaningful features because these weights are tweaked during the training process.

The kernel enables them to perform the operation in high dimensional, implicit feature space without calculating the coordinates of the data in that space. Instead, they compute the inner product of the pictures of all data pairings in feature space. By applying the kernel trick to a linear model, it can be transformed into a non-linear model (Bezdan and Dzakula, 2019).

2.1.4 Pooling

The pooling layer, also known as the down sampling layer, is used to decrease the feature maps dimensionality while retaining the most important data. A filter applies the pooling operation to the input data by sliding over it in the pooling layer (max, min, avg.). In the literature, maximum pooling is most frequently utilized (Alzubaidi et al., 2021).

The essential part of pooling, which is utilized to reduce the complexity of upper layers, is down-sampling. In terms of image processing, it may be comparable to reducing the resolution. Filter count is unaffected by pooling. Max-pooling is one of the most often used pooling methods. The picture is divided into rectangular sub-regions, and only the greatest value discovered inside each sub-region is returned. One of the most prevalent max-pooling sizes is 2 × 2. As shown in Figure 2.2, when pooling is used on the 2-by-2 blocks in the top-left corner, attention is diverted to the top-right corner, and two steps are moved. As a result, stride 2 is used for pooling. It is possible to use stride 1, which is unusual, to prevent down-sampling. Keep in mind that down-sampling does not preserve the position of the data.

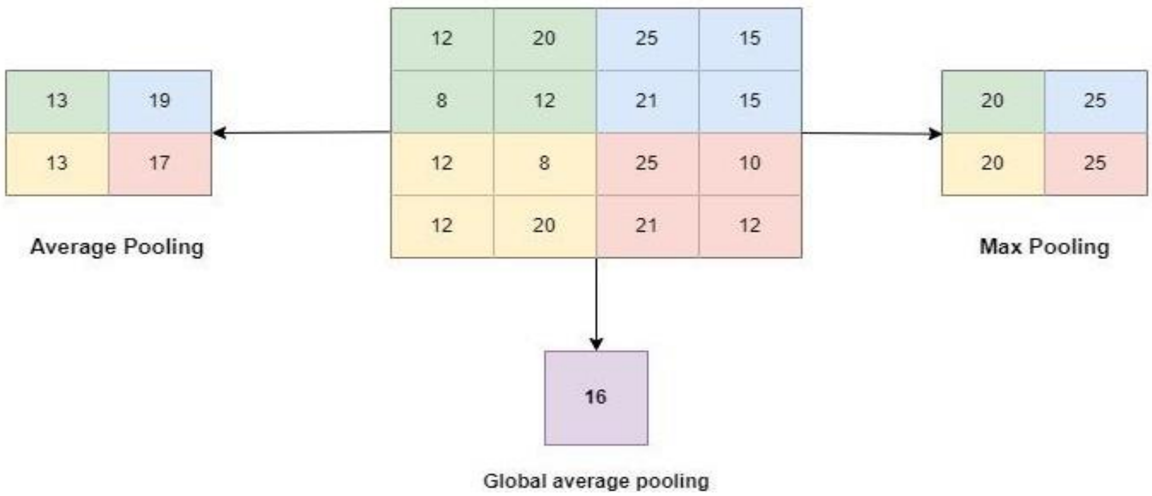


Figure 2.2. Pooling Layer (Source: Mohammed, 2023).

At various pooling levels, various pooling techniques may be applied. Global average pooling (GAP), global max pooling, average pooling, min pooling, and gated pooling are some of these methods. Figure 2.2 depicts each of these three pooling techniques (Du and Swamy, 2019). The primary problem with the pooling layer is that it does not aid CNN in determining whether or not a feature is present in an input image but rather just where that feature is located. Therefore, there are times when CNN’s total rating takes a dip. However, the CNN model leaves out the necessary information.

2.1.5 Non-Linearity (Function of Activation)

The layer of non-linearity follows convolution. Non-linearity allows the generated output to be changed or terminated. This layer is used to restrict or oversaturate the output. Every type of activation function in every type of neural network serves the essential function of mapping input to output. The input value is calculated by calculating the weighted sum of the neuron input and its bias (if present). This indicates that the activation function determines whether or not to fire a neuron in response to a certain input by generating the matching output. In the CNN architecture, non-linear activation layers are used after all layers with weight (also known as learnable layers, such as FC layers and convolutional layers). The mapping of input to output will be non-linear because of the activation layers, non-linear performance, and these layers also enable the CNN to learn extremely complex things (Zhang et al.,2019).

In addition, the capacity to differentiate is a crucial requirement for the activation function since it enables the use of error back-propagation to train the network. The most popular activation functions in CNNs and other deep neural networks are the ones listed below:

a.) Sigmoid: This activation function only allows output values 0 and 1 and accepts real numbers as input (Bhatt et al., 2021).

b.) Tanh: It is comparable to the sigmoid function in that it accepts real numbers as input, but its output range is only between one and one.

c.) ReLU is the most popular function in the CNN context. All of the input values are converted to the positive range. ReLU's primary benefit over other algorithms is the time and resources it saves when used.

For a long time, the Tanh and sigmoid non-linearities were the most prevalent. Non-linearities come in many forms, and they are shown in Figure 2.3. For these reasons, however, the rectified linear unit (ReLU) has seen a surge in popularity in recent years.

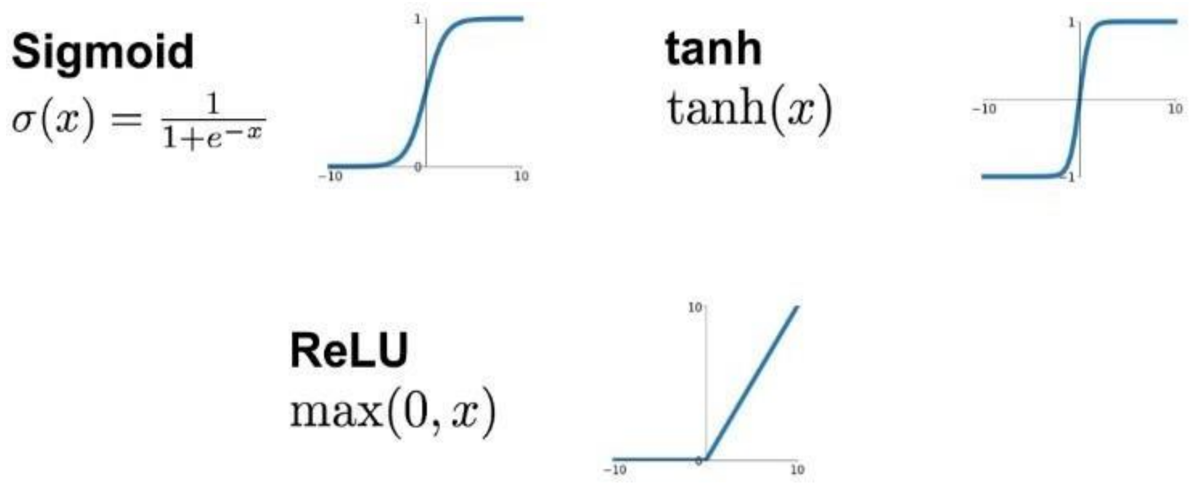


Figure 2.3: Function of Activation (Source: Mohammed, 2023. Theoretical Understanding of Convolutional Neural Network)

Function and gradient definitions using ReLU are simpler. Saturated functions, such as the sigmoid and the tanh, have issues with back-propagation. This phenomenon, known as the “vanishing gradient,” occurs when the gradientsignal gradually decreases as the depth of the neural network architecture grows. This happens because the gradient of these functions is essentially zero on all sides of the center. Nonetheless, the ReLU has a constant gradient for the positive input. While the function cannot be distinguished, it can be ignored during implementation (Alzubaidi et al., 2021).

Third, the ReLU generates a sparser representation because a complete zero is produced by a gradient zero. For sigmoid and tanh, the gradient outcomes are never zero, which may be counterproductive during training (Prakash et al., 2021).

When using ReLU, a few significant problems may occasionally arise. Consider a method for error back-propagation with a greater gradient flowing through it, for instance. The weights will be updated by passing this gradient through the ReLU function in a way that ensures that the neuron will not be stimulated again. This problem is known as “Dying ReLU”. In order to address these problems, there are some ReLU substitutes. These are some of them, as discussed below;

Leaky ReLU: This activation function makes sure that the negative inputs are never disregarded, as opposed to the negative inputs being down scaled by ReLU. It is used to address the Dying ReLU issue.

2.1.6 Fully Connected Layer

Neurons are organized into groups in the fully-connected layer that are reminiscent ofthose seen in traditional neural networks. As shown in Figure 2.4, any node in a layer that isentirely linked is, therefore, directly connected to every node in the layer above and belowit. Figure 2.4 shows that every node in the pooling layer’s most recent frames is connectedas a vector from the fully-connected layer to the top layer. These are the most often utilized CNN parameters within these layers; however, (Zhang et al., 2019). The biggest drawback of a fully connected layer is the large number of parameters that necessitate laborious calculation in training samples. Consequently, we try to minimize the number of connections and nodes.

The eliminated nodes and connections can be satisfiedusing the dropout approach. LeNet and AlexNet, for example, developed a vast and deep network while preserving a constant computational complexity (Du and Swamy, 2019).

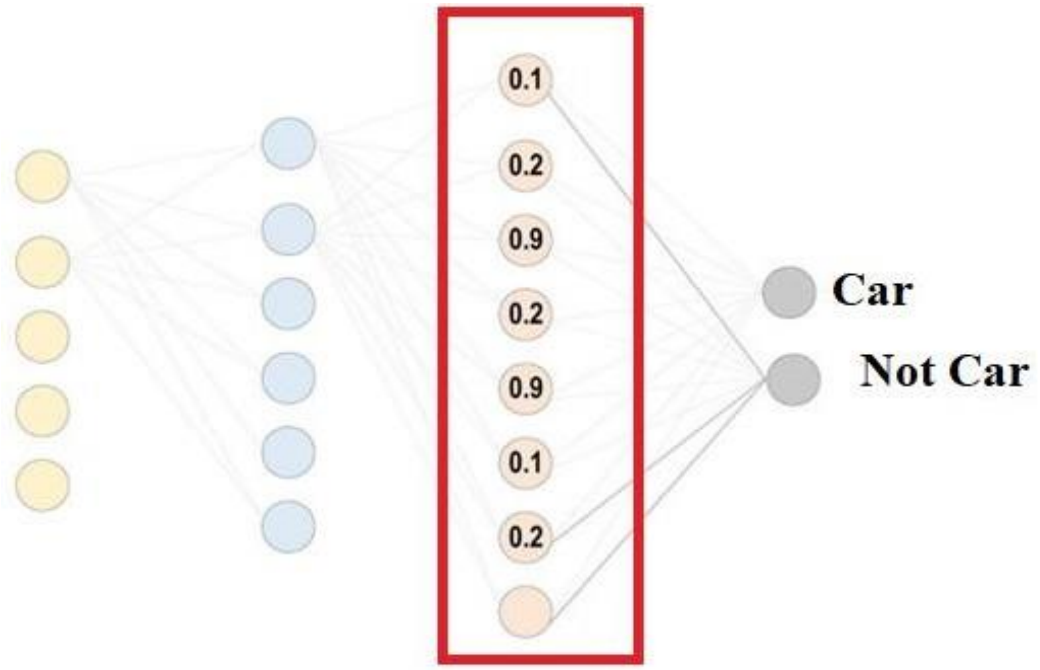


Figure 2.4: Fully-connected layer(source: Mohammed, 2023. Theoretical Understanding of Convolutional Neural Network)

The convolution, which is the core element of the CNN network, is exposed when the non-linearity and pooling layer is added. The three that are most commonly utilized in architecture are as follows:

i. To rephrase, in a completely connected layer, all of the neurons communicate with their counterparts in the layer below. It is a classifier used by CNN.

ii. Being a feed-forward ANN, it performs similarly to a regular multilayer perceptron network. Input to the FC layer comes from the last pooling or convolutional layer.

iii. This is a vector input created by increasing the thickness of the feature maps (Zhang et al., 2019).

Figure 2.4 displays that the FC layer's output is consistent with the final CNN output. The preceding part discussed the various types of layers used in the CNN design; this section will focus on loss functions. Furthermore, the final classification is achieved by employing the output layer, the very last layer of the CNN architecture. A few loss functions are used in the CNN model's output layer to compute the predict error across the training data. As a result of this mistake, the disparity between actual and predicted output is highlighted. Then, it will be improved using the CNN learning approach.

The loss function, however, takes advantage of two inputs to pinpoint the source of the mistake. For CNN, the first parameter is the forecast or estimated output. The second input is the desired output or label. There are many different kinds of loss functions used for different sorts of problems (Bhatt et al., 2021). Below is basic explanation of the many kinds of loss functions:

Training: A training dataset made up of a collection of images and labels (classes,bounding boxes, and masks) is used to train a CNN model.

Back-propagation is a CNN training procedure that measures an error value using the output value of the previous layer. Each neuron's weight in that layer is updated using the error value (Prakash et al., 2021). In order to measure an incorrect value and revise the old weights, fresh weights are employed, as shown in Figure 2.5.

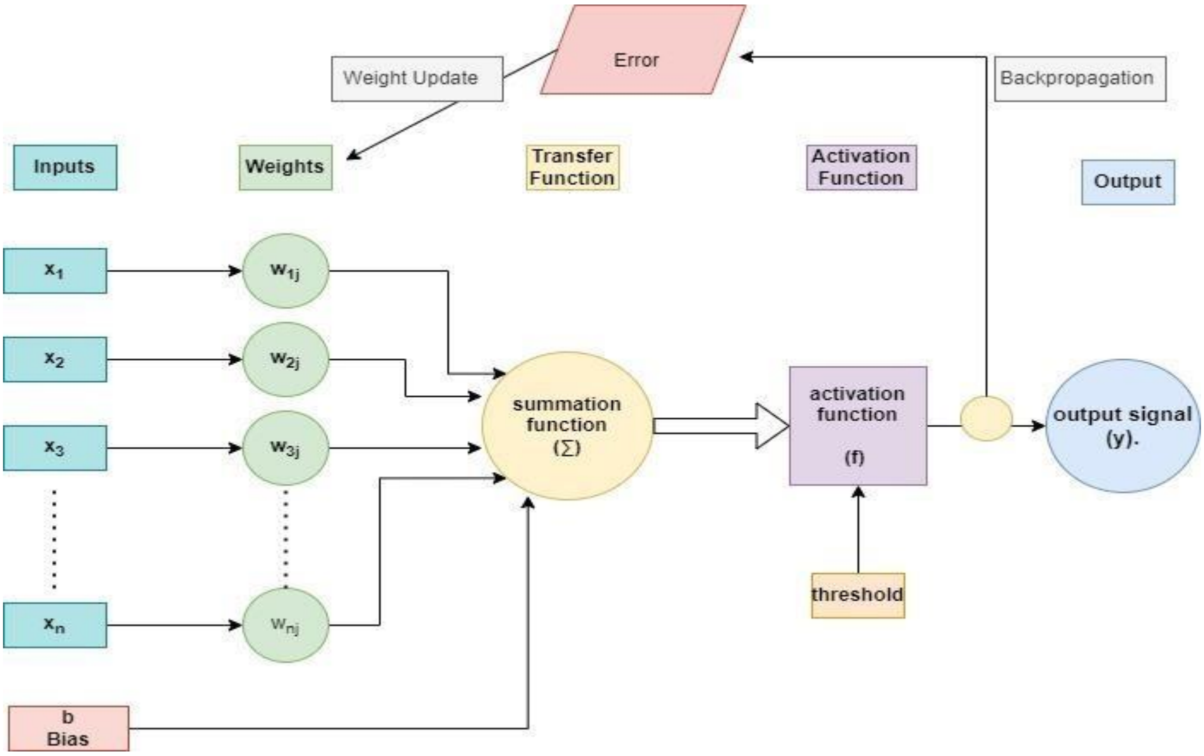


Figure 2.5: Forwardand back-propagation in hidden CNN layers (source: Mohammed, 2023. Theoretical Understanding of Convolutional Neural Network)

Until it reaches the first layer, the algorithm repeats the procedure. All inputs, including the bias unit, are summarized by the activation unit, then, use the activation function to compute the result. The network will then calculate the cost function and send the error back to update the weights unit the cost is minimized.

2.1.7 Advantages of Convolutional Neural Networks over other classical Neural Networks in the context of computer vision

i. One of the main reason for considering CNN is such case is the weight sharing feature of CNN, that reduce the number of trainable parameters in the network, which helped the model to avoid overfitting and as well as to improve generalization.

ii. In CNN, the classification layer and the feature extraction layers learn together, that makes the output of the model more organized and makes the output more dependent to the extracted features.

iii. The implementation of a large network is more difficult by using other types of neural networks rather than using Convolutional Neural Networks.

Nowadays CNN has emerged as a mechanism for achieving promising result in various computer vision based applications like image classification, object detection, face detection, speech recognition, vehicle recognition, facial expression recognition, text recognition and many more (Zhang, 2018).

2.1.8 Application Areas of CNNs

In this section, we discuss some of the major application areas that apply CNN to achieve state of the art performance including image classification, text recognition, action recognition, object detection, human pose estimation, image captioning, etc (Nwankpa et al., 2018).

i.) Image Classification

Because of several capabilities like weight sharing, different level of feature extraction like classifiers, etc, the CNN has been achieving better classificaton accuracy compared to other methods especially in the case of large scale datasets. The first breakthrough in image classification came with the development of AlexNet in 2012, which won the ILSVRC (Image Net Large Scale Visual Recognition Challenge) in that same year. After that, several improvements in CNN model have made by the researchers over the times, and that makes CNN as the first choice for image classification problem.

ii.) Text Recognition

The text detection and text recognition inside an image has been widely studied for a long time. The first breakthrough contribution of CNN in this field begins with LeNet-5, which recognized the data in MNIST dataset with a good accuracy. After that in recent years, with several improvements, CNN contributes a vital role to recognize the text (digits, alphabet and symbols belonging from several languages) inside the image.

iii.) Action Recognition

Based on the visual appearance and motion dynamics of any human body, various effective CNN base methods are now able to predict the action or behavior of human subjects with a notable accuracy. This leads the CNN to the next level in the context of AI. It includes recognition of action from a video sequence or from the still images.

iv.) Image Caption Generation

It means to obtain a description about the target image, which includes detection and recognition of different objects inside that image with their status description. Here we useCNN to perform the first task and we used several Natural Language Processing (NLP) techniques for a textual status description.

v.) Medical Image Analysis

With the advancement in CNN-based image analysis, CNN is rapidly proved to be a stateof-the-art foundation, by achieving enhanced performances in the diagnosis of diseases by processing medical images like MRI, X-rays, etc. Nowadays, CNN based models can successfully diagnose the various health problems like breast cancer, pneumonia, brain tumour, diabetes, parkinson's diseases and many others.

vi.) Security and Surveillance

Nowadays, Security system with Computer Vision capabilities provides constant surveillance to houses, metro stations, roads, schools, hospitals, and many other places that gives the ability to find or identify the criminals even in crowded areas.

vii) Automatic colorization of image and style transfer

In the last few years, with the deep learning revolution, some popular CNN models give an automation way to convert black and white images or gray images to equivalent colorful RGB images. As a result now we can see the old black and white movies in color format. On the other hand image style transfer is a concepts of representing an image in the style of other image, for that a new artificial image could be generated. This style transfer could be efficiently done using convolutional neural network.

viii.) Satellite Imagery

Nowadays, CNN contribute a vital role to detect different natural hazards like tsunamis, hurricanes, floods, and landslides. By satellite image analysis we can do smart city plan, roadway and river extraction, land classification, crop pattern classification, prevention of deforestation and many more.

2.2 Quality Inspection of Screws

During the production process of material components, cracks and scratches can occur. To prevent defective products from getting into the market a quality control after production is necessary. One of the difficulties of these inspections is the parts' complex structure, resulting in various work piece defects (Han et al., 2019)

Many manufacturers use manual inspection to detect defects in the quality control process which is time consuming and affected by the energy level and the worker's experience (Zhou et al., 2019). Quality control in the fastener industry is often carried out by measuring with calipers or optical amplifiers. Some screw manufacturers use human resources to do manual inspections (Zhang et al., 2019). Trained workers are required during this process, which is costly in terms of time and labour (Song et al., 2018). Also, the energy levels of employees are narrowed, which can cause reduced efficiency on long working hours. Often, not every single product is inspected, and samples are used for quality control, resulting in defective parts leaving the manufacturing process (Rezaei-Malek et al., 2019). Mistakes of manual methods then lead to the loss of economic efficiency.

AVI is a quality control technique that utilizes cameras connected to computer systems. The AVI system enhances the quality of the produced goods. Therefore, the inspection checks whether a product differs from specific production rules and specifications. The system captures images of the parts while the conveyor belt is moving. The images are filtered, objects are recognized and features are extracted. Then the detected non-conforming parts are separated. For this method, CNNs are promising because they are able to perform feature extraction and defect recognition on a single network. Hence, the preprocessing of images for certain applications is not necessary. Also, the transferability of the results is facilitated (Konrad andLohmann, 2019).

2.2.1 Quality Control and Machine Learning

A more effective method is automated defect detection in production processes. For detecting surface defects, diverse image processing techniques have been developed (Wang, Chen, 2018). An overview of the relevant related work is shown in Table 1. Recently, in the field of surface detection, various detection techniques based on image processing have been developed. Zhang (2014) designed a product defect recognition system based on machine vision, where an image acquisition module is used to obtain an image of the product, which is processed to judge the degree of defect. This method requires high-quality image, and it is difficult to detect micro defects. An image recognition system to detect screw internal thread was developed by Chen (2014) based on adaptive threshold segmentation and morphological opening calculation to achieve screw identification. However, strong image interference may cause serious under-segmentation, leading to poor detection accuracy. Yan (2013) developed screw thread detection system-based on a CCD (Charged-Coupled Device) digital image correction technique, where maximum variance segmentation and relative sample standard were used to achieve screw identification. However, large image interference can lead to serious over-segmentation, resulting in failure of the inspection. Li and Ren (2012) designed a vision inspection system to capture railway roads images and extract defects from projected contours. Feng et al. (2014) developed an automatic defect detection method using probabilistic topic models. Mariono et al. (2007) used a multilayer perceptual neural classifier to detect missing hex bolts. Aytekin used a high-speed laser range finder, pixel information, and histogram similarity analysis to achieve real-time railway fastener detection (Aytekin and Rezaeitabar, 2015). Prasanna et al. (2012) classified crack images by extracting the curve in the image and using SVM (support vector machine) with hand crafted feature descriptors. Later, Prasanna et al. (2016) combined AdaBoost with random forest method to improve the classifier of Aytenkin. In Zhao et al.(2015), based on the selection of the feature vector of defect images, BP neural networks and SVM were used for pattern recognition, but this method is not robust in extracting feature vectors, and the recognition accuracy is not high. Marco Leo et al. (1998), proposed a system for automatic monitoring of welding process in dry stainless steel kegs for food storage. Cropped are processed by three different algorithmic procedures that perform the monitoring of welding dimensions (spatial metrology), radiometric appearance (radiometric metrology), and local shape analysis, in order to detect thin/thick penetrations, darker areas, and outgrowths respectively.

Often, manufacturers work with inspection systems like high-speed inspection machine using a conveyor and vision cameras to detect surface defects (Iyshwerya andJanani, 2013). Challenges of visual defect detection with images are the greatly heterogeneous appearances of the object surfaces and the defective areas, as well as different rotations and angles of the parts in the inspection process (Arsovskia andCheoka, 2019). All parameters can influence the visual defect detection results, whereas a robust solution is CNN (Hu et al., 2018).

Song et al. (2018), developed a deep CNN-based method to detect micro defects of metal screw surfaces while first locating the screw surface. The data set contained 3,000 samples of defect-free screws, stripped screws, surface-damaged and surface-dirty screws. The model achieved an accuracy of 98.40% and showed the superiority of the DCNN-based approach compared to traditional template matching methods and the LeNet-5-based method. The limitation of their work is that only images of mounted screws with top view were considered. So, this approach is only suitable for special applications like ready-made products with mounted screws.

One study used a detection method based on CNN with automatic image feature extraction to detect surface defects in industry production. The study of Wu, Cao (2019), was able to achieve 90.00% accuracy for screw images with a CNN approach, completed with a voting mechanism for final classification and location. CNN was applied to detect defects in texture images, special structural images such as metallic gasket and screw images. The result was an adaptable model to different data sets. An accurate real-time defect detection system for tiny parts was developed for manufacturing using an end-to-end CNN algorithm. The data set includes four different defect classes of 0.8 cm darning needle and one class of defect-free needles. They use side-view images of the darning needles on a conveyor belt taken from an industrial camera. The different accuracies refer to the different defect types: 98.00%, 99.00%, 97.80%, and 79.40%, so each defect corresponds to a separate model. In contrast to this study, our approach aims to detect all screw defects. The separate models limit the suitability of this approach for realistic quality inspection process. Bergmann (2019) proposed a method for an automated and real-time image classification for different defect types on screws. A variety of defects such as scratches, dents, structural defects like distorted objects, and the absence of parts were considered. Several approaches were tested to improve the results. AnGan, L2 Autoencoder (AE), SSIM AE, CNN feature dictionary, and variation model were considered. The best result was achieved with L2 AE with ratio of 98.00% for correctly classified defect-free screws and 39.00% for defect screws. This corresponds to an accuracy of 83.37%. For accurate quality control in the industrial field, the accuracy is a very important indicator and could be considered as a limitation for this model.

Another anomaly detection approach on the MVTec Anomaly Detection data set of Bergmann (2019) used a technique to visually explain variational AEs via gradient-based attention. By using attention maps, they can localize anomalies in images. They reported the area under the receiver operating characteristic curve (ROC-AUC) with 0.97 for the screw data set (Liu, 2020). Furthermore, a third approach on the same data set was made by Bergmann (2019). They present a student-teacher framework for unsupervised anomaly detection as well as pixel-precise anomaly segmentation. They report the normalized area under the PRO-curve of 0.928 for the screw anomaly detection. The best result achieved by those anomaly detection approaches is a ROC AUC of 0.97, which is shown in Table 2.1.

Table 2.1: Approaches for Automated Defect Detection of Screws.

REFERENCE	METHOD	PERFORMANCE
Song et al. (2018)	DCNN	98.40%
Wu et al. (2019)	CNN	90.00%
Yang et al. (2019)	CNN	79.40%-99.00%
Bergmann et al. (2019)	L2 AE	83.37%
Liu (2020)	Variational AE	AUC-ROC: 0.97

Song (2018) applied a deep CNN-based method to detect micro defects of metal screw surfaces from a top view of mounted screws. An adaptable model was created by Wu (2017) by using a CNN approach to detect defects in texture images, special structural images such as metallic gasket and screw images. Only one defect type was taken into account for this approach. Defect darning needles can be detected using several single models with the approach by Yang (2019), which is not practical in an industrial process.

2.2.2 Internet of Things (IoT)

Although the internet of things (IoT) technique has been proposed for a period of time, it has not yet been widely adapted in the manufacturing industry. Integrating AI and internet of things (IoT) techniques into automated factories has turned into a trend in recent times (Chettri and Bera, 2019). The apparent difference between the smart factory and the traditional automated factory is whether IoT technology has been introduced or not. IoT systems are comprised of intelligent terminal equipment, wireless networks, cloud, and big data management. Considering the limitation of the devices, the IoT technology transfers big data from cameras or mechanical devices embedded with sensors and software to the cloud platform through the network (Roman et al., 2018).

Therefore, data clustering is utilized to handle big data. Factories can access big data that are stored in the cloud efficiently and quickly. Recently, several studies have introduced the IoT technique to improve the industrial problem, such as fault diagnosis, insulator string defect detection (Song et al., 2021), and LCD display defect detection(Variz et al., 2019). A smart factory using IoT techniques can manage automation equipment and automated defect detection devices with more intelligence than automated factories, which can significantly improve product quality and production efficiency. The current screw factory manufacturing process mainly includes screw production, defect detection, and product packaging.

A major concern for the screw factory is how to minimize defects and prevent the flow of defective products. The screw is designed to be fastened into position within a hole by means of the thread surrounding the flank surface, which is beneficial for fasteners since they cannot fall out and damage the machinery. Screws must comply with a strict quality and safety requirements. Critical application with regard to high precision, stability, and safety are other important elements for selecting screws. Therefore, the task of detecting defective screws plays an important role in the producing screws.

2.3 Deep Learning

Nowadays, deep learning (DL) technology is considered as one of the hot topics within the area of machine learning, artificial intelligence as well as data science and analytics, due to its learning capabilities from the given data (Sarker, 2021). Many corporations including Google, Microsoft, Nokia, etc., study it actively as it can provide significant results in different classification and regression problems and datasets. In terms of working domain, DL is considered as a subset of machine learning (ML) and AI, and thus DL can be seen as an AI function that mimics the human brain's processing of data. The worldwide popularity of "Deep learning" is increasing day by day, based on the historical data collected from Google trends.

Deep learning differs from standard machine learning in terms of efficiency as the volume of data increases, discussed briefly in section "Why Deep Learning in Today's Research and Application?." DL technology uses multiple layers to represent the abstraction of data to build computational models. While deep learning takes a long time to train a model due to a large number of parameters, it takes a short amount of time to run during testing a compared to other machine learning algorithms (Shrestha and Mahmood, 2019).While today's Fourth Industrial Revolution (4IR or Industry 4.0) is typically focusing on technology-driven "automation, smart and intelligent systems", DL technology, which is originated from ANN, has become one of the core technologies to achieve the goal. A typical neural network is mainly composed of many simple, connected processing elements or processors called neurons, each of which generates a series of real-valued activation for the target outcome.

2.3.1 Research and Application of Deep Learning

Deep learning approaches have grown dramatically in terms of performance in a wide range of applications considering security technologies, particularly, as an excellent solution for uncovering complex architecture in high-dimensional data (Cintra et al., 2018). Thus, DL techniques can play a key role in building intelligent data-driven systems according to today's needs, because of their excellent learning capabilities from historical data. Consequently, DL can change the world as well as humans' everyday life through its automation power and learning from experience. DL technology is therefore relevant to artificial intelligence, machine learning and data science with advanced analytics that are well-known areas in computer science, particularly, today's intelligent computing. In the following, we first discuss regarding the position of deep learning in AI, or how DL technology is related to these areas of computing (Shretha, 2019).

2.4 Summary of Reviewed Literature

With the rapid development of technology, artificial intelligence techniques have achieved impressive success and turned into a hot topic in image processing research. Utilizing deep learning techniques to solve the defect detection issue can alleviate the need for complicated manual feature extraction. Deep learning techniques can automatically learn and extract meaningful features from raw images more comprehensively than previously possible. Although deep learning technology can automatically extract features in a better way than the traditional manual methods of feature selection, this kind of supervised learning based on deep learning networks needs a large amount of labeled data for model training (Krizhevsky et al., 2017). It is difficult for traditional methods to accurately identify screw surface defects. In this work, method for identifying micro defects of metal screw surfaces was developed based on Deep Convolutional Neural Network and an optical platform for acquiring screw images was built. Images of defected and defect-free screw surfaces were collected, which were used to train the designed Deep Convolutional Neural Network (DCNN). To enable efficient detection, we first locate screw surfaces in the pictures captured by the cameras, so that the images of screw surfaces can be extracted, which are then input to CNN-based defect detector. The proposed method does not need to acquire the features of the screw surface images in advance, and robust to illumination changes. Comparisons with traditional machine vision techniques, e.g., template matching-based techniques, demonstrate the superiority of the proposed deep CNN-based one. (Bengio et al., 2016).

Table 2.2: The comparison between supervised learning and unsupervised learning.

Approach	**Methods**	**Strengths**	**Weakness**
Supervised-based classifier	Classification network: CNN.	Fast training and Inference.	**1**. Requires numerous abnormal and normal images for training. **2.** Hard to collect abnormal images in practical situations. **3**. Tedious manual annotation work.
	Semantic network: U-Net, FCN.	Fast training and Inference.	
	Object detection network: YOLO, Faster R-CNN.	Identify the defect with the bounding boxes.	
Unsupervised-based classifier	Adversarial Autoencoder	Model training without annotation; requires only positive datasets for model training.	**1.** Imprecise defect localization with poor reconstruction. **2.** A large amount of clean normal data are needed to obtain useful results.

2.5 Research Gap

Although a good number of work have been done using different Convolutional Neural Network (CNN) techniques, no work has been done using remodeled the VGG16 CNN network architecture first modelled by Simonyan and Zissermann(2015), which is remodeled using Google's machine learning architecture, Tensorflow 2.0, and Keras 2.3.1 on countersunk head 5.2mm threaded screw of a production company to detect defects such as surface dirt, surface damage, and free defect of the proposed screw, afterward, the performance of the proposed CNN were compare with other developed CNN models.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Equipments/Apparatus

Figure 3.1 shows the optical experimental platform system for acquiring screw images. We use an industrial camera RS-A2300-GC50 (manufactured by China Microview, Beijing, China) with a CMOS resolution of 1600 × 1200 pixels and a 16 mm (M0814-MP2) lens. The distance between the camera and the object is about 200 mm. The light source controller controls the brightness of the DOME light source. The large opening angle of the DOME light source is helpful for uneven surface imaging, and multiple reflections through the inner wall of the hemisphere can completely eliminate shadows, which is helpful for metal or mirror surface inspection. The captured image is a 24-bit image of size 1600×1200 pixels in BMP format. The length of a pixel in the image is approximately 0.0765 mm (i.e., 122.35 mm/1600 pixels = 0.0765 mm/pixel).

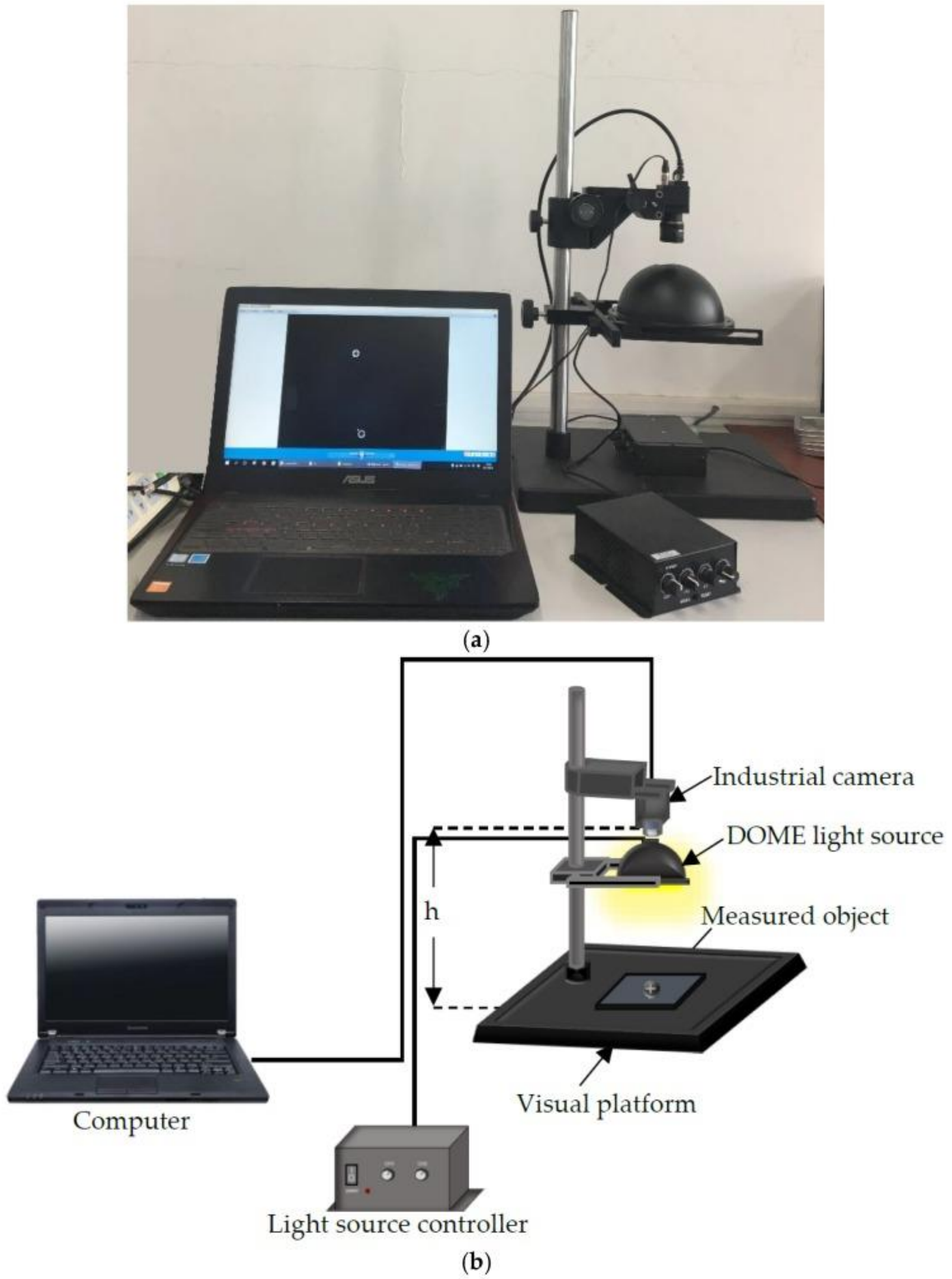


Figure 3.1: Experimental System for image acquisition.

(a) Detection System,

(b) System Structure diagram.

Table 3.1: Equipment.

ITEM	SPECIFICATION/APPLICATION
Industrial Camera	RS-A2300-GC50 (manufactured by China Microview, Beijing, China) with a CMOS resolution of 1600 × 1200 pixels and a 16 mm (M0814-MP2) lens. Used to show the optical experimental platform system for acquiring screw images.
Metal Screws	GB819 cross counter-sunk head screw M3s with a diameter of 5.2 mm. Used as the subject matter of the research.
Personal Computer (ASUS)	ASUS notebook ROG GX501 VIK7700, CPU Model=Intel Core i7 7700HQ, Core/thread number= Four core/eight threads, Memory capacity = 16 GB, Hard drive capacity = 1 TB, Graphics chip = NVIDIA GeForce GTX 1080 Max-Q, and Video memory = 8 GB. Used as The computer for the experiment which aided in the CNN process that involved application of Google's machine learning architecture Tensorflow, and Keras software.
High-Resolution Industrial Camera	RGB sensor with 2,048×2,048 pixels. Used to view the defect images on the surface of the screw.

3.2 Data Preparation

The metal screws used in this study are GB819 cross countersunk head screw M3s with a diameter of 5.2 mm. The types of surface defects considered in this work include dirt on surface, surface damage, and stripped screws. An example of the image contour query, and the region of interest is selected to obtain the screw images show in Figure 2b where the sample size is 96 × 96, which is captured under different illuminations. The training sample set consists of 230 defect-free screws, 287 stripped screws, 205 surface-damaged screws, and 256 surface dirty screws. A large number of extended samples are generated through translation and distortion. The expanded sample set has a total of 3000 samples. Different lighting conditions are achieved by adjusting the light source controller. Figure 3.3 shows the effect of adjusting the intensity, rotation, and distortion of the same screw image. Some randomly-selected defect-free samples are shown in Figure 3.4, and some samples of defective screws are shown in Figures 3.5, 3.6 and 3.7

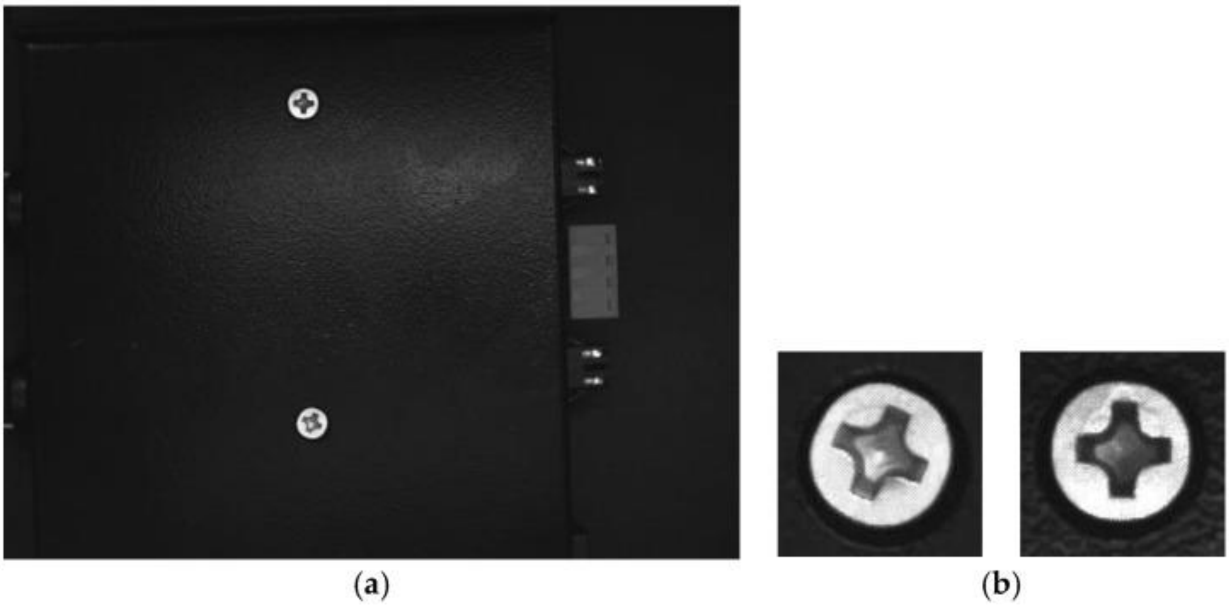


Figure 3.2: Data acquisition.

(a) Original image,

(b) Detected screws.

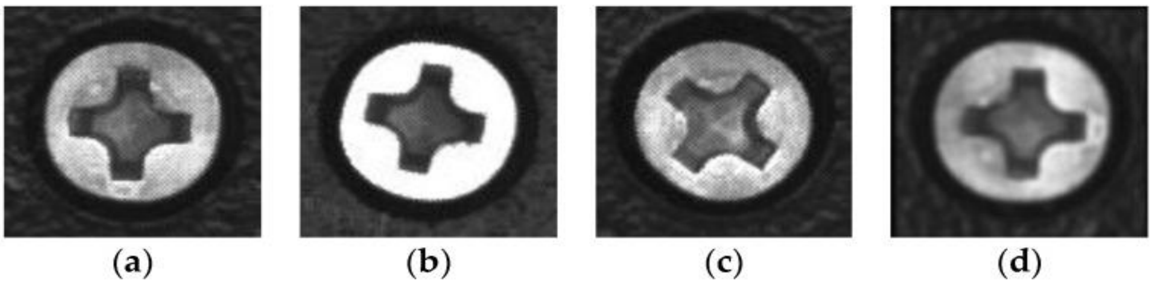


Figure 3.3: Data enhancement. (a) Original image, (b) image obtained by adjusting the light intensity, (c) rotated image, (d) distorted image.

Figure 3.4: Defect-free screws.

Figure 3.5: Stripped screws.

Figure 3.6: Surface-damaged screws.

Figure 3.7: Surface dirty screws.

3.3 Detection Method

To efficiently detect the defects on screw surfaces, the first step is locate the screw surfaces in the images (as shown in Figure 3.9) captured by the industrial camera. Then, the screw surface images are extracted, which are fed to the trained deep CNN for defect detection. The screw surface defect detection method is summarized as follows:

1. Use the optical platform to collect the object image, which may contain multiple screw surfaces, as show in Figure 3.9.
2. Carry out gray-scale processing, which turns the three channel colour images into single-channel gray-level image;
3. The gray image of screw is binarized, i.e., 0-255 gray image is converted into 0 (black) or 255 (white) image, as shown in Figure 3.10;
4. Through image contour query (Suzuki and Be, 1985), get the contours of the screw surfaces, as shown in Figure 3.11.
5. Obtain the positions, heights and widths of the screw surfaces in the image based on their contours. Take the screw in Figure 3.11 as an example. In Figure 3.11a, points A, B, C, and D,are the leftmost, uppermost, lowermost, and rightmost points of the screw respectively. Then, the position (x, y), the height h, and the width w of the screw are obtained, i.e., $x = x1$, $y = y1$, $h = |y2 - y3|$, $w = |x4 - x1|$ as shown in Figure 3.11b;

6. Based on the position, height and width of each screw surface, extract the colour images of the screw surface from the original image captured in Step 1. The size of the image is adjusted to 32 × 32, and the image is then input to the trained CNN for defect detection;

7. The screw positions obtained in Step 5 are marked on the original image, and the defect types are also indicated with different colour borders. Figure 3.13 shows an example.

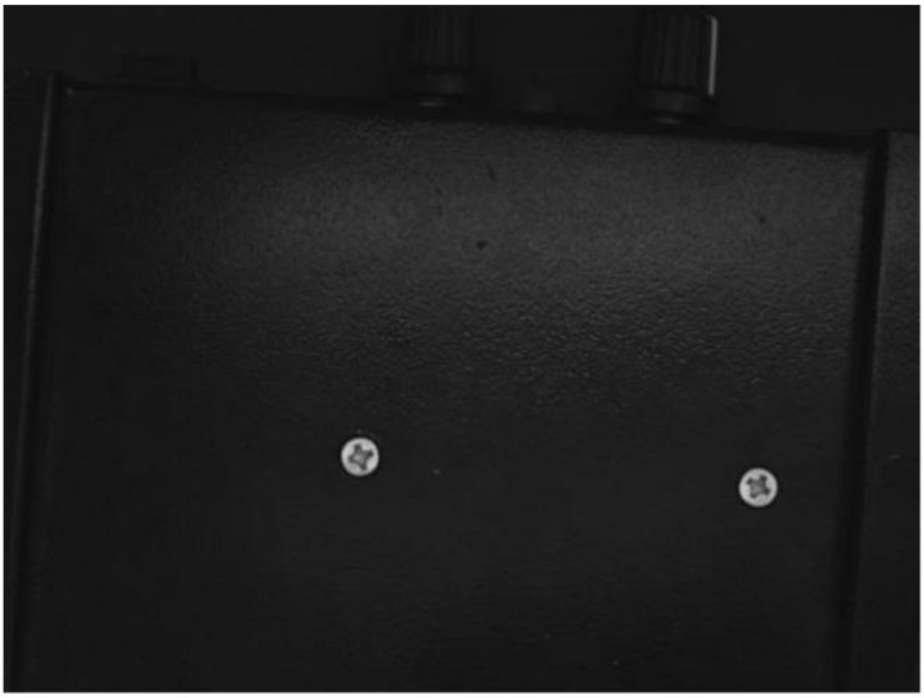


Figure 3.8: Original Image Captured by the Camera.

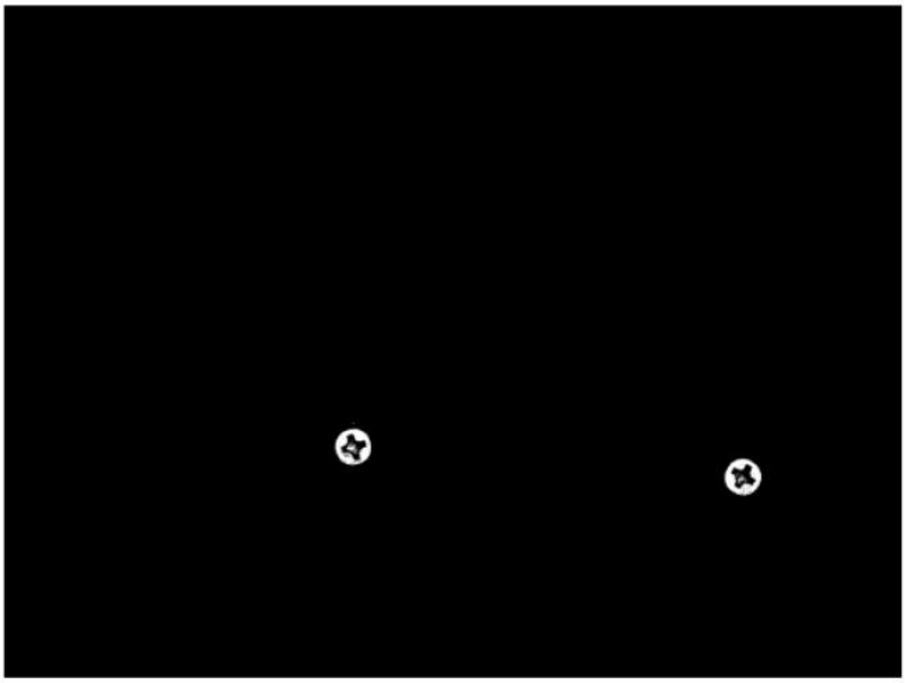
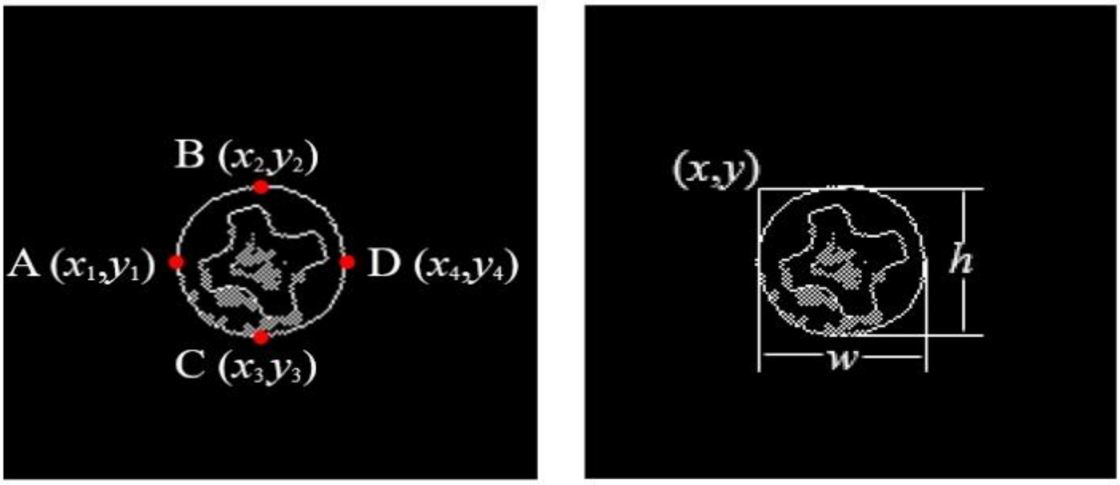


Figure 3.9: Binarized Image

Figure 3.10: Contours Query



(a) (b)

Figure 3.11: Screw Surface Localization.

(a) Boundary points,

(b) Located screw surface area

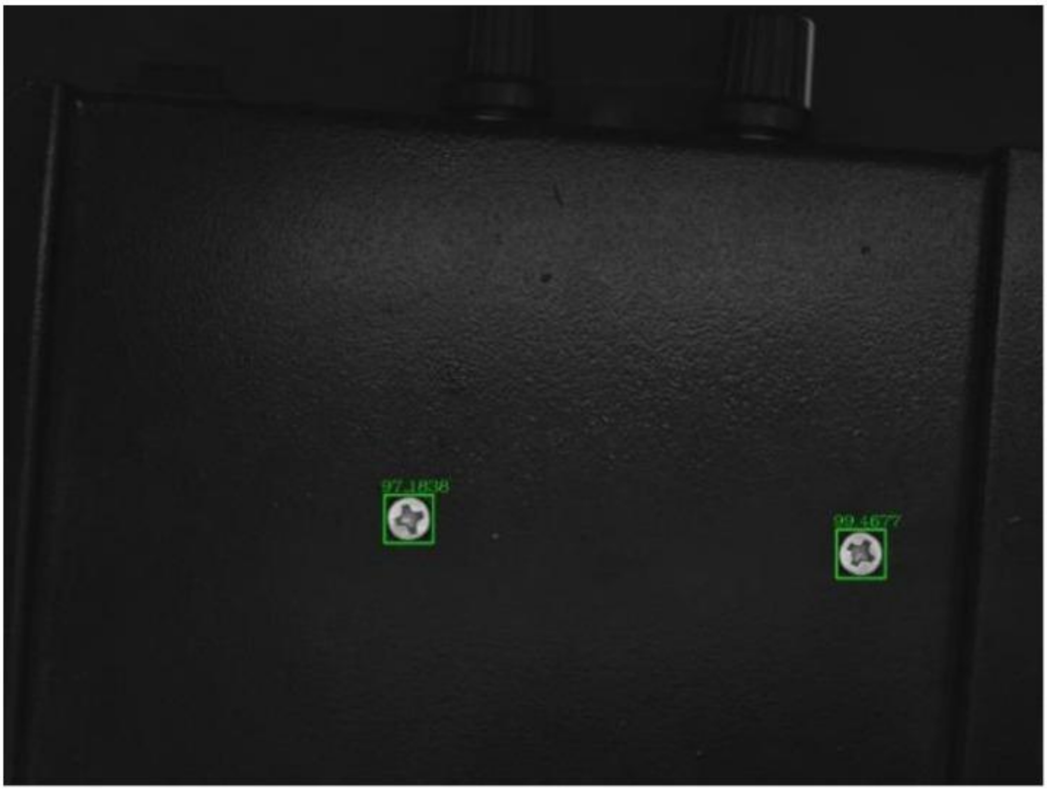


Figure 3.12: Detection Results.

3.4 Neural Network Structure

In this work, we propose the use of convolutional neural networks (CNN) to design the detection method. The essence of CNN is to use the huge data to filter the image multiple times through and reduce the dimension by down-sampling. In addition, the nonlinear fitting of the activation function is used to obtain more abstract and deeper essential features of the target, so as to realize the recognition of the target and solve the important problem of manual design features in the past. Therefore, a deep learning model based on convolutional neural network is suitable for image processing and other related machine learning problems.

The metal screw surface defect detection method in this paper is developed based on the traditional LeNet-5 (Lecun et al., 2015). The architecture of the network is shown in Figure 3.8. The first layer is the input layer with size $32 \times 32 \times 3$ (i.e., the length and width of the images are 32, and the number of color channels is 3). The input data pass through the architecture and are generalized with spatial size reduction to $4 \times 4 \times 64$ at Pool 3; the output of the layer is then fed into rectified linear unit (ReLU) layer. Finally, the softmax layer outputs the probabilities of the four cases: defect-free screw, dirty screw, damaged screw, and stripped screw. A dropout layer is located after each layer of Fc1, Fc2, and Fc3. In order to keep the dimension unchanged after convolution, zero padding is used to get better results in the final feature search without affecting the operation speed.

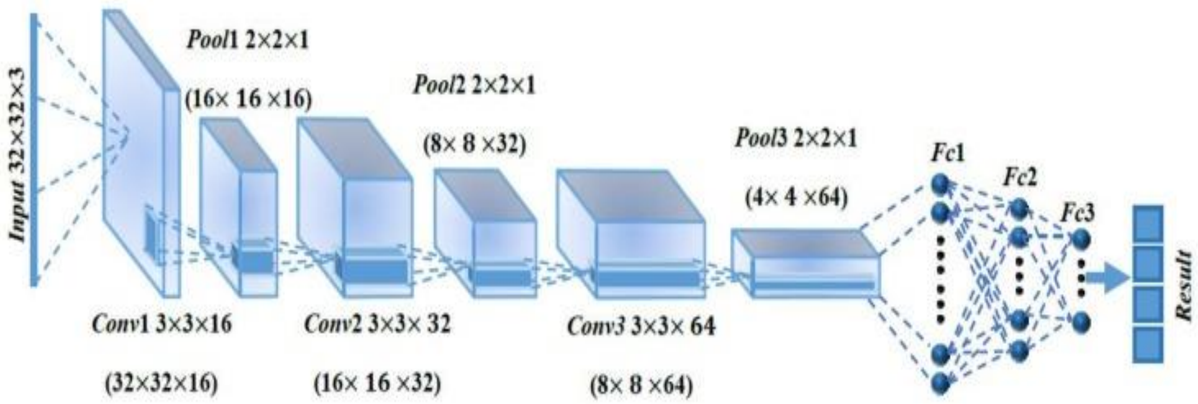


Figure 3.14: Structure of the CNN used in this work.

3.5 Experiments

Determining appropriate hyperparameters (e.g., learning rate and regularization parameters) is cumbersome, and there is no accurate guidance for optimizing those parameters. Therefore, these parameters are obtained through trial and error, guided by checking the verification set error (Bengio et al., 2016). In order to demonstrate the superiority of the CNN proposed in this work, it is compared with the traditional LeNet-5. The computer used in the experiment is ASUS notebook ROG GX501 VIK7700 with the configuration shown in Table 3.2. The CNN trained the model with tensorflow and Google's machine learning architecture.

Table 3.2: Computer configuration.

CPU Model	Intel Core i7 7700HQ
Core/thread number	Four core/eight threads
Memory capacity	16 GB
Hard drive capacity	1 TB
Graphics chip	NVIDIA GeForce GTX 1080 Max-Q
Video memory	8 GB

3.6 Model Architecture

We chose the VGG16 architecture because of its ability to solve complicated objected detection problems, and because of its representation depth, it is suitable for the accuracy of classification. We adjusted the VGG16 based model to achieve the best result. As a result of the depth and number of fully connected nodes, the architecture requires much memory, but this does not pose a problem as powerful hardware is available for industrial inspection and manufacturing (Baz, 2014). The architecture consists of five blocks of convolutional layers. They all have kernel size of 3×3. After every block, a max pooling 2D is applied to sample down the input. By using pooling layer, the resolution of the feature maps is decreased allowing a spatial invariance to be generated. We used two dense layers before the classifier, one with a filter size of 128 and the other one with 32. In addition, two dropout layers were applied, both with a rate of 0.5, to prevent the model from over-fitting. For the layers, the activation function rectified linear unit (ReLU) was used. With layer weights, regularizers penalties are imposed on the layer parameters. Using the kernel regularizer L2, penalties are applied to the layer's kernel. The regularizer was set to the default value of 12=0.01 (Tao et al., 2014). For the output layer, which is dense layer, the sigmoid activation function was applied, which is commonly used for binary classification as it returns a value in the range of 0 to 1. The filter size of the output layer is 1.

For the training, the layers of our network were frozen so that the weights do not differ. The Adam optimizer (Kingma and Ba, 2015) is applied because the optimizer has the benefits of two other optimizers, which are the RMSprop and Adagrad (Kingma and Ba, 2015). Our architecture has a total of 14,784,513 parameters. After the training, the model can be optimized with Fine Tuning methods. For our model, we unfroze the last ten layers to make small adjustments to improve the performance. Furthermore, the Adam optimizer was set to a learning rate of 5×10-5. The training and Fine Tuning were done with 200 epochs and a batch size of 32. Subsequently to the Fine Tuning, the model can be evaluated by calculating several performance indicators such as balanced accuracy, kappa and precision of the model and providing a confusion matrix.

3.7 Convolutional Neural Network and Transfer Learning

A widely used Deep Learning method is CNN. In the past, CNNs already achieved outstanding results in the detection, segmentation and recognition of objects and areas in images (Krizhevsky et al., 2017). This approach is also cost-efficient, robust and fast. CNNs are used to convert data that is in the form of multiple arrays, such as a colour picture consisting of 2D arrays containing pixel intensities in the three-colour channels.

A CNN is organized into several stages. These stages consist of different convolutional and pooling layers. The units of convolutional layers are organized into feature maps. Each unit is connected to local patches in the feature map of the previous layer by a filter bank. The result of this locally weighted sum is then passed through a non-linearity, e.g., a ReLU (LeCun et al., 2015)

CNNs need a large number of images to be able classify reliably. This has the effect that training as architecture requires a lot of processing capacity and time. Sometimes it is not possible to collect a large quantity of training data. With the help of high-performance learners, it is possible to train a CNN successfully even with only a small amount f data. This technique is called Transfer Learning (Weiss et al., 2016). Transfer learning has already been effectively applied for image classification (Wen et al., 2014).

The training of CNN consists of two steps. First, the relevant structures and shapes are learned, from which more abstract objects can be derived and recognized. In the second step, a fully trained network is established based on the features already learned. Here Transfer Learner is being applied. During this process, Fine Turning is utilized. Here, a transfer of the trained layers only occurs on the output layer. The output layer is adapted by adjusting the number of object classes of the new model to be detected and newly trained (Wang, 2018). Successful implementation of Fine Tuning has already been taken place is several applications (Penatii et al., 2015)

For the CNN architecture, we used the network architecture VGG16. This network was introduced by Simonyan and Zissermann (2015). The architecture consists of a total of 16 weighted layers and is used for large image recognition. The size of the convolution filters is only 3×3, and the depth of the network can be increased. All convolutional layers have the same filter size. The five max pooling layers are in a kernel size of 2×2. The activation of all hidden layers takes place with ReLU. In total, the network consists of 13 convolutional layers and two fully connected layers, each of which has total of 4,096 units. The final layer is activated with softmax and with 1,000 nodes. The default input size for the VGG16 architecture is 224×224×3 (Simonyan et al., 2013).

3.8 Dataset and Preprocessing

The data set used for our training and evaluation of our Deep Learning approach is the MVTec Anomaly Detection data set by (Bergmann et al., 2019). The data set was collect in 2019 and consists of 15 categories, including images of several objects and textures. The data set contains a total of 3,354 images of several objects and textures. Out of the 15 categories, we consider the screw images to detect defective and non-defective parts. The screw images can be categorized into two classes. The first class consists of correct screws, whereas the second class includes screws that have defects on the surface of the objects, like scratches and dents.

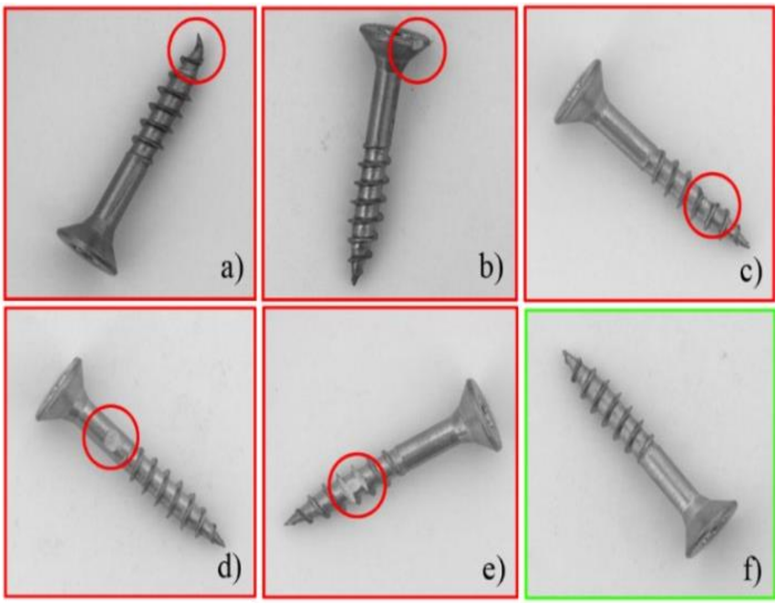


Figure 3.14: Defective screws with defects of the;

a.) Front (b.) head (c.) thread side (d.) neck (e.) thread top (f.) a non-defective screw.

The defects were created manually with the aim to produce realistic anomalies similar to real-world industrial inspection scenarios. The images were acquired with high-resolution industrial RGB sensor with 2,048×2,048 pixels. The screws have different damage types, which can be seen in Figure 3.14. The damage types on the screws include:

a) The screw front.

b) The screw head.

c) The screw thread side.

d) The screw neck.

e) The screw thread top.

Image “f” shows an example of a defect-free screw.

The data set for screws consists of 119 images of defective and 361 images of non-defective screws. For preprocessing, a balanced data set of 119 images for each class has been selected. Therefore, all images of the defect class were used, 119 images out of the non-defective screw class were chosen randomly. Afterward, a split into train, test and validation set was carried out. The data set was split into ratio of 60:20:20. Thus, the number of training images is 142, the number of validation images is n=48, and the number of test images is n=48. The model was trained with the training and validation images. For the calculation of the performance indicators, only the unseen test images were used.

The RGB-colored images were resized to 224×224 pixels and normalized into the range of [-1,1]. Resize and rescale were applied to all images of the balanced data set. Furthermore, Data Augmentation was used to prevent over-fitting and to improve the performance of the classification. Therefore, only the training images were rotated (Chollet, 2017).

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Results

Firstly, we used the traditional method to detect the screw defects. Figure 4.1 is the original image of the object to be detected (the yellow numbers are post-marked to ease the description of the screw surfaces). There are five screws in the image: Screw 1 and 5 are defect-free screws, 2 is a dirty screw, 3 is a surface-damaged screw, and 4 is a striped screw. We selected an image of a defect-free screw as a template, and used various template-matching methods for defect detection, including the normalized correlation matching method (Aoki et al., 1996), normalized correlation coefficient matching method (Deng et al., 2006), correlation coefficient matching method (Liu and Lin, 2011), normalized square difference matching method (Hisham et al., 2016), square difference matching method (Kusuma et al., 2016), and correlation matching method (Liu and Li, 2007). The results are shown in Figure 4.2. It can be seen that the template matching method cannot accurately detect the two defect-free screws.

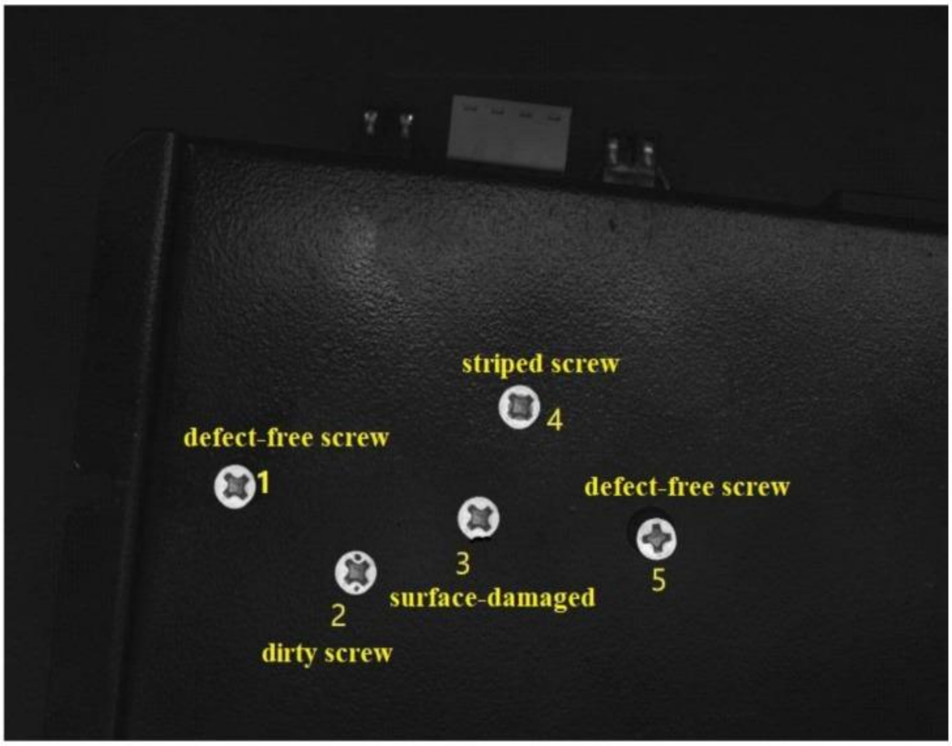
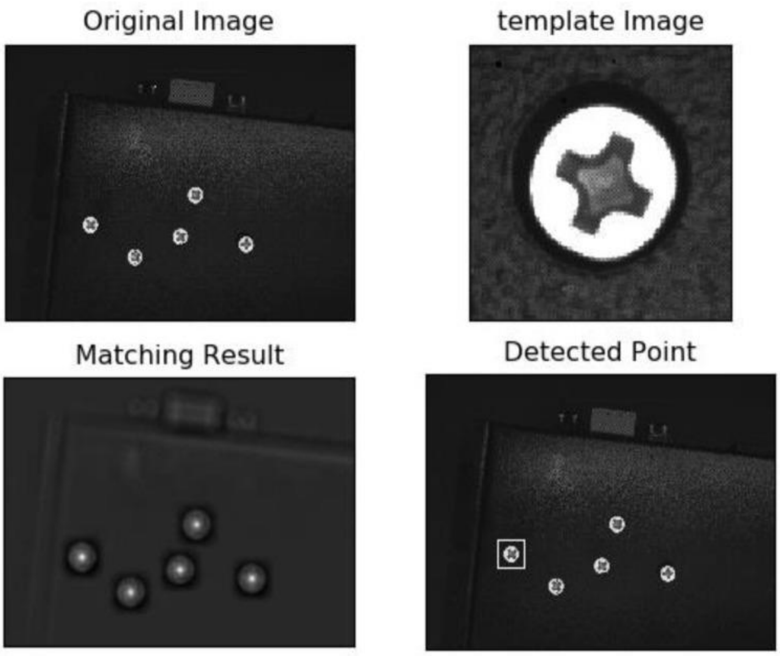


Figure 4.1: Screw for testing.



(a)

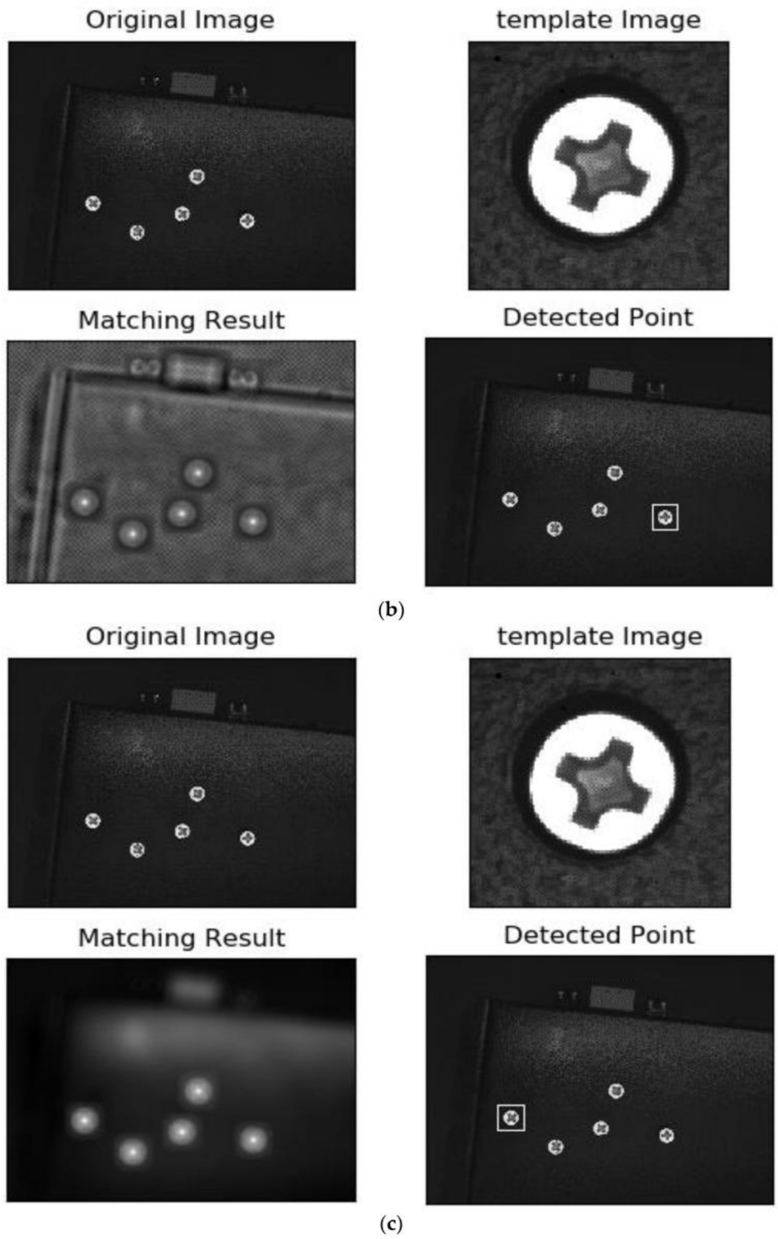
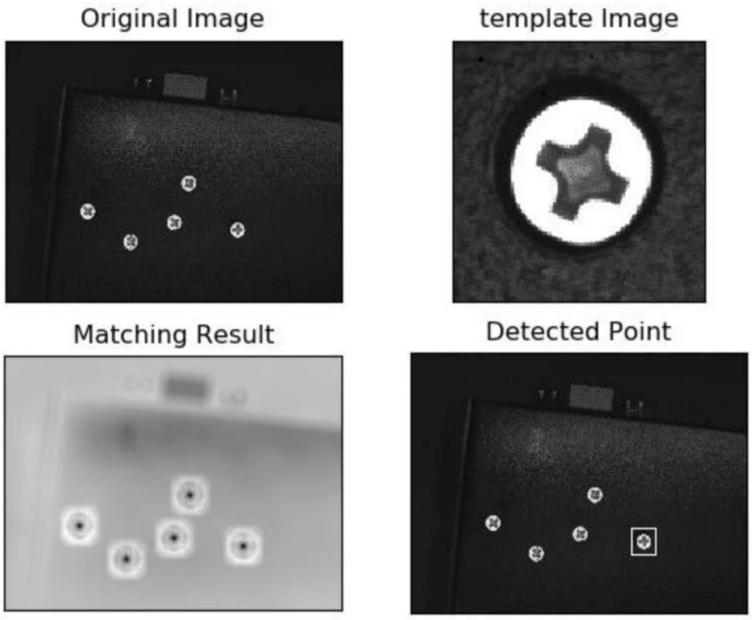
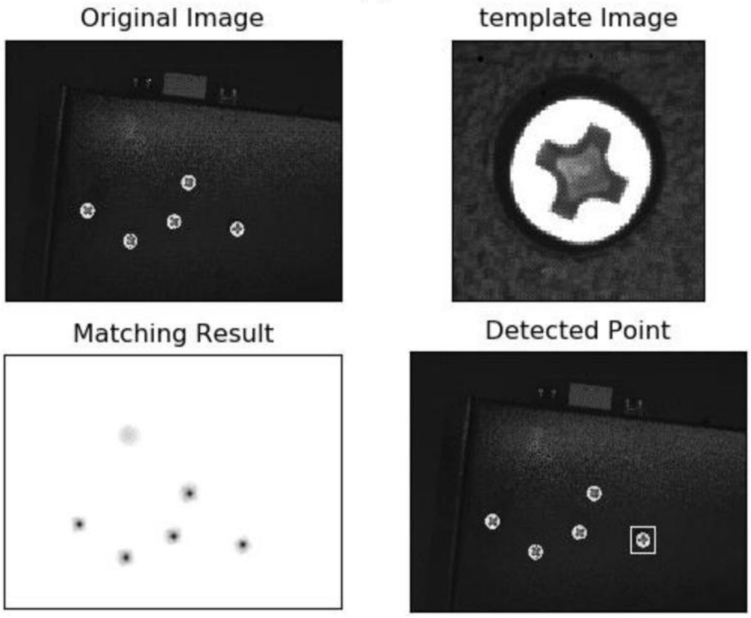


Figure 4.2: Traditional template matching-based detection methods.



(d)



(e)

Figure 4.2: Traditional template matching-based detection methods.

(a) Normalized correlation matching method.

(b) Normalized correlation coefficient matching method.

(c) Correlation coefficient matching method.

(d) Square difference matching method.

(e) Correlation matching method.

In order to further verify the superiority of the proposed model, the proposed CNN was compared with the traditional LeNet-5. The same screw data set is used to train the two networks. Experiments showed that the accuracy and loss tend to be stable after 1000 iterations. Figure 4.3 shows the accuracy of the traditional LeNet-5 and the proposed deep CNN with 1000 iterations. It can be seen that the accuracy of the proposed DCNN is much higher than the traditional LeNet-5 at the beginning of the training, and the accuracy of the training was to 100% with 550 iterations, and about 100% accuracy was achieved with 800 iterations. Figure 4.4 shows the minimum training loss of the traditional LeNet-5 and the proposed DCNN with 1000 iterations. It can be seen from the figure that the loss rate of the proposed DCNN decreases rapidly, which is slightly better than that of the LeNet-5. When the number of iterations is about 550, the loss rate is close to 0. Figure 4.5 shows the detection error.

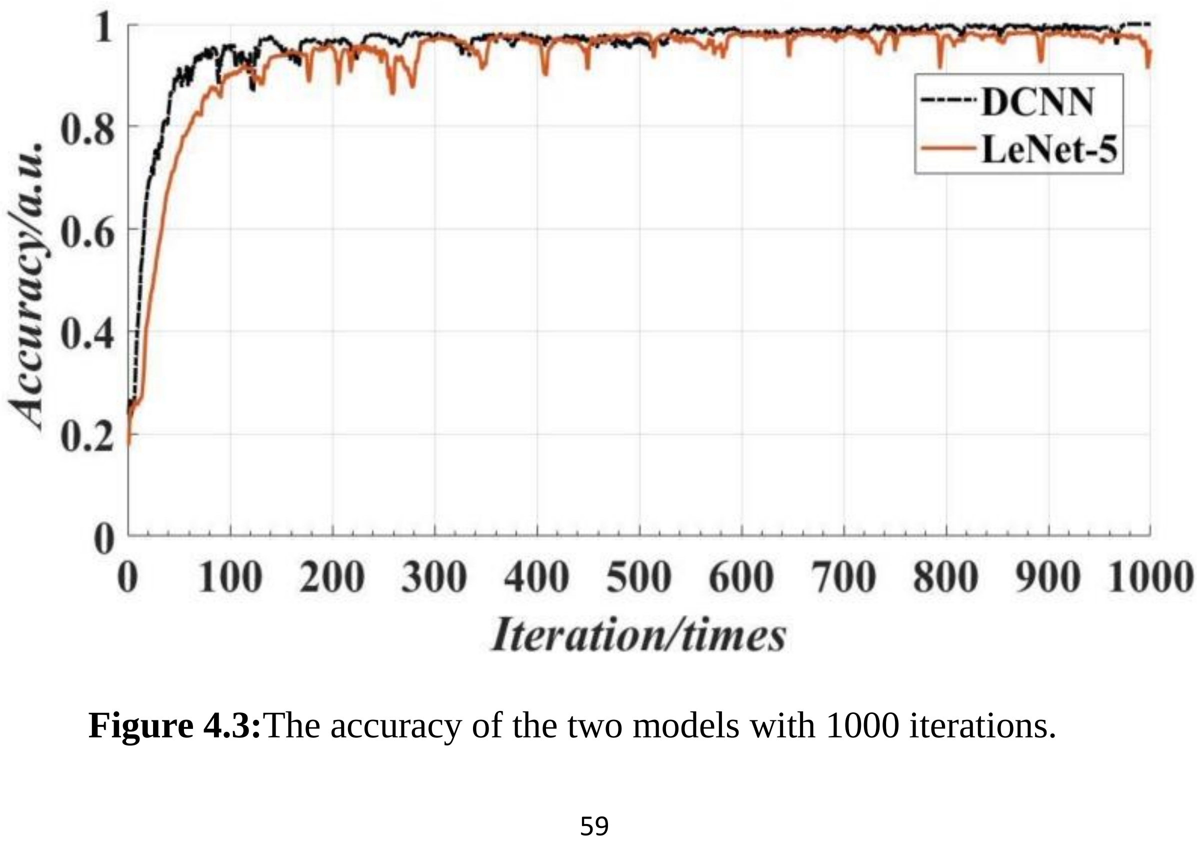


Figure 4.3: The accuracy of the two models with 1000 iterations.

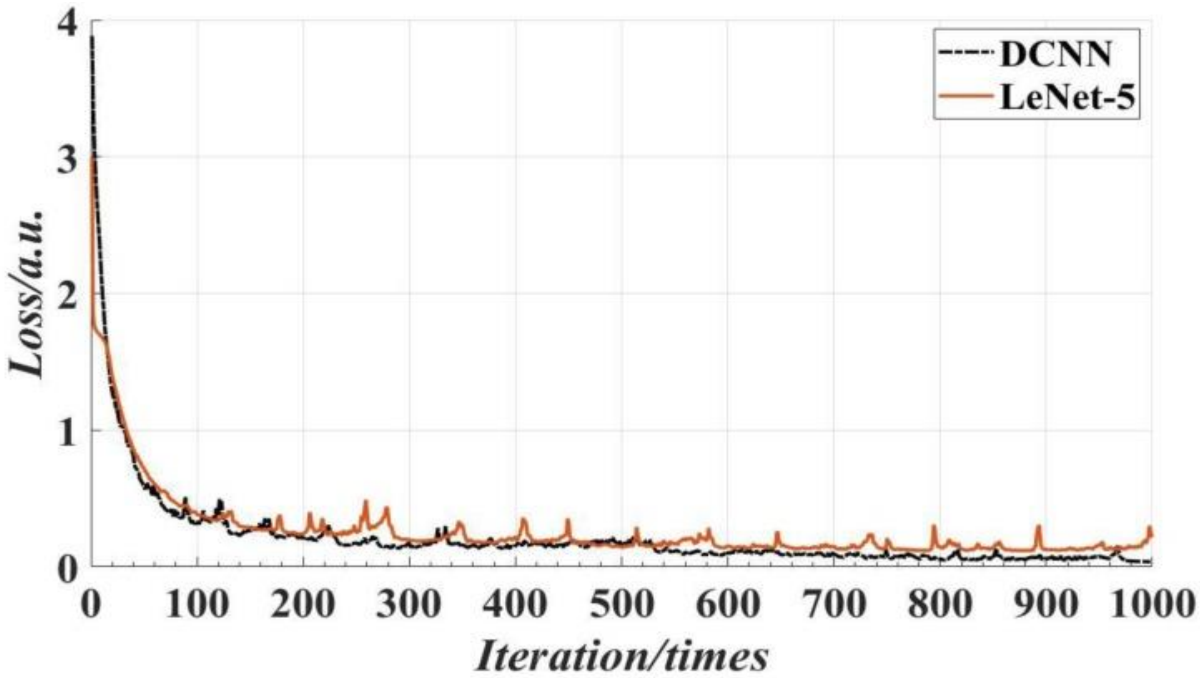


Figure 4.4: The minimum training loss of the two models with 1000 iterations.

Figure 4.5: The test results of two models.

(a) The test results of LeNet-5.

(b) The test results of the proposed DCNN.

To better examine the detect performance of the two networks, the same verification set is used to test the accuracy of the two methods, and

1000different types of images are selected as the test set. The results are shown in Table 4.1

Table 4.1: Recognition rate of different categories of samples in the test set.

	Total Number of Images	**Correct Detection**	**Error Detection**	**Accuracy**
LeNet-5	1000	958	42	95.8%
The proposed DCNN	1000	984	16	98.4%

A comparison of different CNNs is shown in Table 4.2.

Table 4.2: The comparison of different CNN.

	Time	**Accuracy**
YOLO	Faster	Low
R-CNN	Low	Low
Faster-RCNN	Fast	High
SSD	Low	Higher
The proposed DCNN	Faster	Higher

For the training of the CNN, Keras 2.3.1 was used with Tensorflow 2.0 backend. For training of the model, we used the publicly available Google Collaboration service. It uses an Nvidia Tesla K80 12 GB GPU. To be able to detect the defect screw, the model was trained 200 epochs and a batch size of 32. For the Fine Tuning, the learning rate was set to 5×10-5. To evaluate the model, a confusion matrix has been calculated as well as several performance indicators like the balanced accuracy, kappa, precision, recall and the area under the receiver operating characteristic curve (AUC-ROC).

Thus, it was possible to achieve a balanced accuracy of 98%. The model predicted all 24 defective screws right. Unfortunately, one screw was wrongly classified as a defective screw, and 23 were correctly predicted as defect-free screws. Table 4.3 shows the summarized classification results in a confusion matrix.

Table 4.3: Confusion Matrix.

		Reference	
		Defective	***Non-Defective***
Prediction	***Defective***	**50.00% (24)**	2.08% (1)
	Non-Defective	0.00% (0)	**47.92% (23)**

In Table 4.4, the performance indicators are presented. Cohen's kappa has value of 93.96%, and the model reached a true positive rate of 100.00%. The positive predictive value is 96.00% and the AUC-ROC equals 0.990. These results prove a good performance of the model.

Table 4.4: Performance Indicators

Performance Indicator	**Value**
Accuracy	97.92%
True positive rate	100.00%
Positive predictive value	96.00%
Prevalence	50.00%
Balanced Accuracy	97.92%
Kappa	93.96%
AUC-ROC	0.990

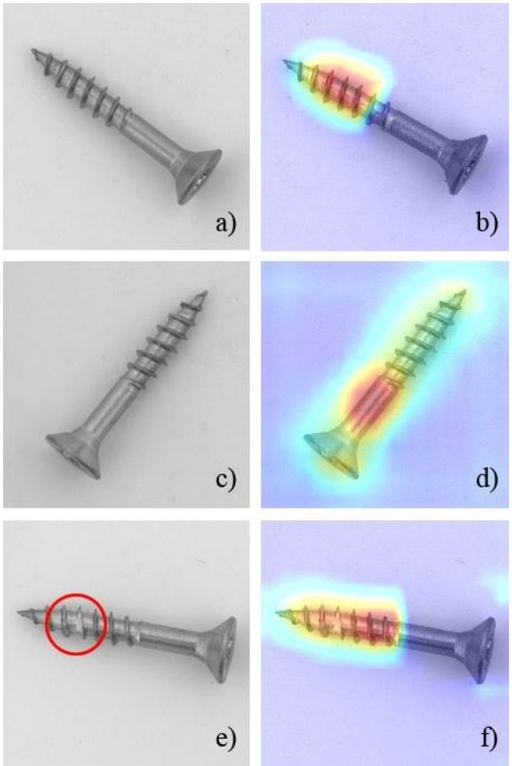


Figure 4.6: Examples of a false classified non-defective screw

a) image, b) heat map; falsely classified as non-defective screw

c)image, d) heat map; correctly classified asnon-defective screw

e) image, f) heatmap; correctly classified as defective screw

To understand why the model falsely predicted one defect-free screw as defective, we used the Grad-CAM heatmap algorithm (Selvaraju et al., 2017) to visualize the important areas. In order to create the heatmaps, the images for the Grad-CAM are up-and downscaled and therefore not pixel accurate (Morbidelli et al., 2020).

The one falsely predicted screw a) image in Figure 4.6, is defect-free but was predicted as defective. The heatmap b) in Figure 4.6, shows that the model mainly focused on the thread of the screw, which is coloured in red. The restof the image, which is coloured in blue, is not predictive for the model. Image c) shows a correctly classified non-defective screw. Heatmap d) shows that the whole screw is recognized, but the focus is mainly on the neck of the screw. A correctly predicted defective screw is shown in image e) in Figure 4.6. The heatmap f) indicates the focus on the thread of the screw, where the defect is located. Thus, the area of the defect is clearly shown.

As shown in Table 4.2, our CNN model performs efficiently and shows excellent classification outcomes. The performance is also shown in the confusion matrix in Table 4.3, supporting the good results. Further, the confusion matrix displays that all types of defective screws were correctly classified, while only one image out of 24 defect-free images is falsely classified.

This might be because some of the screw defects are just slight defects. Furthermore, the lighting can make it appear as if the surface has been scratched. Therefore, the model has probably classified the screw as faulty. The evaluation of the heatmaps shows that the network focuses especially on the parts where the defects are location. This includes the screw neck and thread, as shown in Figure 4.6. The related work shows that a CNN approach is highly promising for the defect detection of screws and tiny metal parts. Also, anomaly detection with an L2 AE approach was used to detect such defects in screw (Bergmann andFauser, 2019). The current studies achieve an accuracy from 79.40% to 99.00% for defect detection. However, those approaches rely on additional imaging techniques or simply focus on the detection of only single defect types (Gross andBreitenbach, 2021). If different defects are considered, several models were needed to detect those with a high level of accuracy (Yang and Li, 2019).

Therefore, with an accuracy of 97.92%, our model, which can detect different defect types, also shows clear advantages compared to models with better performance results like the work by Yang et al., (2019). Even though they considered realistic quality inspection scenarios in their experiment, the four defects could only be detected with separate models. This means that only one defect type could be predicted with a 99.00% accuracy, which might not be practicable in real-world manufacturing. Furthermore, with an accuracy of 98.40%, high performance results were also achieved by Song et al., (2018). However, they only considered very specific perspectives. The top view perspective of mounted screws used in this work is not suitable for the realistic screw production process.

We compared the prediction time of our model on CPU and GPU with different batch sizes to evaluate the suitability of our approach for an application during production. Table 4.5 shows good inference times for CPU as well as for GPU. With an increase of the batch size, the time the computer needs for each step is doubled. This applies to the performance speed of CPU and GPU. A fast performance speed enables the possibility to be used in real-time applications.

Smaller and lighter models are very useful for portable solutions but less suitable for very complex models (Sandler et al., 2018). The lager model VGG16 is able to solve very complex object recognition problems. In industrial inspection and manufacturing, high performance hardware is available, making our approach highly practicable (Simonyan and Zisserman, 2015).

TABLE 4.5: Comparison of Inference Time on CPU and GPU

Batch Size	**CPU**	**GPU**
1	320 ms	24 ms
2	614 ms	44 ms
4	1166 ms	75 ms
8	2104 ms	92 ms
16	3225 ms	167 ms

TABLE 4.6 Performance Indicators of other Transfer Learning Architectures

Architecture	Accuracy
Xecption	83.30%
VGG19	71.67%
InceptionResNetV2	61.65%
ResNet50	55.00%
EfficientNetB3	53.36%
EfficientNetB4	52.81%

4.2. Discussion

This model outperformed several other architectures, which were tested and evaluated for this approach. The pre-trained VGG16 network performed overall with the best results and an accuracy of 97.92%. Only one screw image was falsely predicted as defective. Table 4.6.shows that Xecption architecture performed with an accuracy of 83.30% and achieved the best results of the compared networks. The light Xception architecture is very easy to be defined and modified. Also, architectures such as VGG16 have those properties.Other networks like VGG19 and InceptionResNetV2 performed with an accuracy below 80% and therefore were not considered any further. The ResNet50 architecture did not perform well, and so this approach was not considered in detail. The EfficientNetB3 and

EfficientNetB4 are mobile-sized models and can be scaled up, but in our use-case only achieved an accuracy of under 60%.

To evaluate the performance of the proposed methods, template matching combined with anomaly models of CAE and AAE networks are investigated in this section. Initially, the datasets utilized in this work are described in detail. Then, three experiment parts are presented. Moreover, the template matching method is evaluated first to view the effectiveness of the merged images. Thereafter, the comprehensive results of the two anomaly networks, namely CAE and AAE, are discussed and compared to explore the detection performance. Finally, specific descriptions are provided.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1. Conclusion

In this work, an engineered CNN-based approach for detecting micro defects on metal screw surfaces was developed. Experimental results demonstrated achieved remarkable 98% accuracy, surpassing LeNet-5 and traditional methods.Leveraged pre-trained VGG16 architecture, outperforming previous investigations and showed that a CNN network could be capable of identifying various screw defects (dents, scratches, missing/distorted parts).

Real-world alignment suitable for in-situ industrial defect detection, in addition, the heat maps show high reliability in defect detection. The internal validity of our model is high due to the train-test-validation-spilt, where testing is only carried out on unseen test images (Yadavet al., 2016). All defect parts were predicted correctly. Thus, our model contributes to zero-defect manufacturing. It is not only about the improvement of quality but also the learning process of quality inspection by identifying and analyzing any outliers in the process. Any potential risk for quality has to be considered and not just an improvement of the yield.

Due to the fast performance speed of our architecture for CPU and GPU, it could be possible to run it on a mobile application. This would enable the model in further areas of quality control. Thus, it could be applied in every step of the production and delivery of screws. For instance, to check the screw randomly at the incoming goods inspection. This model is suitable for large number of images and high-speed inspection in industrial environment. The proposed method may also be used in other industrial inspection application, such as bottle cap defect detection, mobile phone screen defect detection, beverages, and more.

5.2 Recommendations

i. Future focus on larger, detailed dataset for enhanced technique assessment and model reliability.

ii. A comprehensive dataset can improve predictions, precise defect recognition, and quality comparison.

iii. Enabling detection of screws with multiple defects is crucial for comprehensive quality control.

iv. Adaptation of the model for mobile devices expands application in screw manufacturing stages.

v. Research stacking multiple CNN architectures for a more robust defect detection model.

vi. Reinforce data augmentation techniques for improved model resilience (blur, zoom, brightness).

vii. Investigate human-computer interaction impact on workforce collaboration and automated quality control.

viii. Analyze user workload and user-oriented concepts for insights in production settings.

5.3 Contribution to Knowledge

i. This study significantly advances the field by investigating the application of artificial neural networks and optimization techniques in screw production.

ii. The research highlights the essential role of defect detection in maintaining production quality and cost efficiency.

iii. The utilization of deep convolutional neural networks (DCNN), combined with advanced image acquisition methods, demonstrates a substantial leap in micro flaw detection on metal screw surfaces.

iv. The study's findings showcase the superiority of the proposed DCNN technique over traditional methods, offering a faster processing time and higher accuracy in defect detection.

v. This work provides valuable insights into enhancing quality control measures and production processes within industrial contexts, contributing to improved manufacturing practices and cost management.

REFERENCES

Alaluf, A. and Birnbaum, D. (2013). Inspection of PCBs by laser-induced fluorescence. Circuit World, vol. 28, pp. 21-28.

Alzubaidi, L., Zhang,J., Humaidi A.J., Al-Dujaili, A., Duan Y., Al-Shamma, O., Santamara, J., Fadhel, M.A., Al-Amidie, M. andFarhan, L., (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, and future directions. J.Big Data, pp. 8, 83.

Aoki, T., Nagatomo, K. and Suzuki, T. (1996). Development of square groove seam tracking method using image processing with normalized correlation pattern matching for one-side submerged arc welding. Jpn. Weld. Soc. 58, pp. 84-85.

Arlot, S. andCelisse, A. (2010). A Survey of Cross-Validation Procedures for Model Selection. Statistics Surveys, vol. 4, pp. 40-79.

Arsovskia, S., Cheoka, A.D., and Yatb, C.Y. (2019). Automated Visual Inspection Model for Screw Detection on the Moving Objects in Industrial Quality Control. ISM ‘19 Procedure, pp. 1-21.

Aytekin, C., Rezaeitabar Y., Dogru, S. and Ulusoy, I. (2015). Railway Fastener Inspection by Real-Time Machine Vision. IEEE Transactions on Systems Man Cybernetics-Systems. Vol. 45, pp. 1101-1107.

Azizpour, H., Razavian, A.S., Sullivan, j., Maki, A., and Carlsson, S. (2015). From Generic to Specific Deep Representations for Visual Recognition. IEEE CVPRW ‘15 Procedure, pp. 36-45.

Batmaz, Z., Yurekli, A., Bilge, A. and Kaleli, C. (2019). A review on deep learning for recommender systems: Challenges and remedies. Artif. Intell. Rev. pp. 52, 137.

Baz, D.E. (2014). IoT and the Need for High Performance Computing. IIKI ’14 Procedure, pp. 1-6.

Bengio, Y., and Grandvalet, Y. (2004). No Unbiased Estimator of the Variance of K-Fold Cross-Validation. Journal of Machine Learning Research, vol. 5, pp. 1089-1105.

Bengio, Y., Goodfellow I.J. andCourville, A. (2016). A Deep Learning, An MIT Press Book. Available online: http://www.deeplearningbook.org

Bergmann, P., Fauser, M., Sattlegger, D., and Steger, C. (2019). MVTec AD – A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. IEEE/CVF CVPR Procedure, pp. 9584-9592.

Bergmann, P., Fauser M., Sattlegger, D., and Steger. (2020). Uninformed Students: Student-Teacher Anomaly Detection with Discriminative Latent Embeddings. IEEE/CVF CVPR ’20 Procedure, 2020, pp. 4183-4192.

Bezdan, T. andDzakula, N.B. (2019). Convolutional Neural Network Layers and Architectures. In International Scientific Conference on Information Technology and Data Related Research; Singidunum University: Belgrade, Serbia, pp. 445-451.

Bhatt, D., Patel, C., Talsania, H., Patel, J.,Vaghela, R., Pandya, S., Modi, K. andGhayvat, H. (2021). CNN variants for computer vision: Histroy, architecture, application, challenges and future scope. Electronics, pp. 10, 2470.

Buettner, R. (2013). Cognitive Workload of Humans Using Artificial Intelligence Systems: Towards Objective Measurement Applying Eye-Tracking Technology. KI ‘13 Procedure, vol. 8077 of LNCS, 2013, pp. 37-48.

Buettner, K., Sauer, S., Maier, C. and Eckhardt, A. (2018). Real-Time Prediction of User Performance based on Pupillary Assessment via Eye Tracking. AIS Transactions on Human-Computer Interactions, vol. 10, no. 1 pp. 26-56.

Canziani, A., Culurciello E., and Paszke, A., (2017). Evaluation of Neural Network Architectures for Embedded Systems. IEEE ISCAS '17 Procedure, pp. 1-4.

Chen, J., Liu Z., Wang, H., Nunez, A., and Han, Z., (2018). Automatic Defect Detection of Fasteners on the Catenary Support Device Using Deep Convolutional Neural Network. IEEE Transactions on Instrumentation and Measurement, vol. 67, no. 2, pp. 257-269.

Chen, Y.Q., Zhu, X.J., Wang, Y.H., Guo J.L. and Chen, L.Q., (2014). Research on Image Recognition of Internal Thread. Tool Eng. vol.12, pp.77-79.

Chollet, F., (2017). Xception: Deep Learning with Depthwise Separable Convolution. IEEE CVPR '17 Procedure, pp. 1800-1807.

Choi, S., Cho, O., and Kim, E., (2015) Design of Automated Screw Blade Welding Defect Detection System for Image Processing –based Zero-defect Mass-production. CSCI '15 Procedure, pp. 845-846.

Cintra, R.J., Duffner, S., Garcia, C. and Leite, A., (2018). Low-complexity approximate Convolutional Neural Networks. IEEE Trans. Neural Network Learning System.Vol. 29, pp. 5981-5992.

Deng, H.H., Yang, L. and Huang, S.T., (2006). The Successive Elimination Algorithm in the Image Matching Based on Normalized Cross-Correlation Coefficient. Microelectronics Computer. 23, pp. 227-230.

Dong, K., Lin C., Chen, C and Sze, J., (2018). The Internet technology for defect detection systems with deep learning method in smart factory. In Proceedings of the 2018 4th International Conference on Information Management (ICIM), Oxford, UK, 25-27 May 2018; pp. 98-102.

Du, K.L. and Swamy, M.N.S., (2019). Neural networks and statistical learning, second edition. In Neural Networks and Statistical Learning, 2nd edition; Springer: Berlin/Heidelberg, Germany, pp. 1-988.

Fang, X., (2017). Understanding deep learning via back-tracking and deconvolution. J.Big Data, pp. 4, 40.

Feng, H., Jiang, Z., Xie F., Yang, P., Shi, J. and Chen, L., (2014). Automatic Fastener Classification and Defect Detection in Vision-Based Railway Detection Systems. IEEE Transactions on Instrumentation and Measurement. vol. 63 pp. 877-888.

Gabler, U., Osterreicher, I., BoskP., and Nowak, C., (2007). Zero Defect Manufacturing as a Challenge for Advanced Failure analysis. IEEE/SEMI ASMC '07 Proc., pp. 341-344.

Gadelmawla, E.S., (2017). Computer Vision Algorithms for Measurement and Inspection of External Screw Threads.Measurement., vol. 100, pp. 36-49.

Gaertner, M., Sauter, D., Baumgartl, H., Rieg T., and Buettner, R., (2021). Multi-Class Emotion Recognition within the Valence-Arousal-Dominance Space Using EEG. AMCIS '12 Proc., pp. 1-10.

Gross, K.A., Lungevics, J., Zavickis, J., and Pluduma, L., (2018). A Comparison of Quality Control Methods for Scratch Detection on Polished Metal Surfaces. Measurements, vol. 117, pp. 397-402.

Gross, J., Breitenbach J., Baumgartl, H., and Buettner, R., (2021). High Performance Detection of Conrneal Ulceration Using Image

Classification with Convolutional Neural Networks. HICSS-54 Procedure, pp. 1-10.

Han M., Wu Q., and Zeng, X., (2019). Single-Scale Workpiece Defect Detection Based on Deedp Learning. 12th International Congress on Image and Signal Processing. CISP-BMEI '19 Procedure, pp. 1-5.

He, K., Zhang, X., Ren, S., and Sun, J., (2016). Deep Residual Learning for Image Recognition. IEEE CVPR '16 Proc., pp. 770-778.

He, X., Chen Y., and Ghamisi, P., (2020). Heterogeneous Transfer Learning for Hyperpectral Image Classification Based on Convolution Neural Network. IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 5, pp. 3246-3263.

Hisham, M.B., Yaakob, S.N., Raof, R.A.A., Nazren, A.B.A. and Embedded, N.M.W., (2016). Template Matching Using Sum of Squared Difference and Normalized Cross Correlation. Research and Development. pp. 100-104.

Iorga, C., and Neagoe, V.E., (2019). A Deep CNN Approach with Transfer Learning for Image Recognition. ECAI '19 Proc., pp. 1-6.

Iyshwerya, K., Janani, B., Krithika, S., and Manikandan, T., (2013). Defect Detection Algorithm for High Speed Inspection in Machine Vision. IEEE ICSSS '13 Proc., pp.103-107.

Kingma, D., and Ba, J., (2015). Adam: A Method for Stochastic Optimization. ICLR '15 Proc., pp. 1-15.

Konard, T., Lohmann, L., and Abel, D., (2019). Surface Defect Detection for Automated Inspection Systems Using Convolution Neural Networks, in Mediterranean Conference on Control and Automation (MED) '19 Proc., pp. 75-80.

Korzh, O., Cook G., Andersen, T., and Serra, E., (2017). Stacking Approach for CNN Transfer Learning Ensemble for Remote Sensing Imagery, in Intelligent Systems Conference '17 Proc., pp. 599-608.

Koushik, J., (2016). Understanding Convolutional Neural Networks. Available online: http://arxiv.org/abs/1605.09081

Kusuma, B.A., Nugroho, H.A., Wibirama, S., (2016). Spinal Curvature Determination from Scoliosis X-ray Image using Sum of Squared Difference Template Matching. In Proceedings of the International Conference on Science and Technology, Yogyakarta, Indonesia, pp. 27-28, 29-34.

Krizhevsky, A., Sutskever, I., and Hinton, G.E, (2017). ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems, vol. 60, no. 6, pp. 84-90.

LeCun, Y., Bengio, Y., and Hinton, G., (2015). Deep Learning. Nature, vol. 521, no. 7553, pp. 436-444.

Lehr, J., Schluter, M., and Kruger, J., (2019). Classification of Similar Objects of Different Sizes Using a Reference Object by Means of Convolutional Neural Networks, in ETFA '19 Proc., pp. 1519-1522.

Leo, M., Del Coco, M., Carcagni P., Spagnolo, P., Mazzeo, P.L., Distante, C. and Zecca, R., (2018). Automatic Visual Monitoring of Welding Procedure in Stainless Steel Kegs. Optics and Lasers in Engineering. Vol. 104, pp. 220-231.

Li, Q. and Ren, S., (2012). A Real-Time Visual Detection System for Discrete

Surface Defects of Rail Heads. IEEE Transactions on Instrumentation and Measurement. vol. 61, pp. 2189-2199.

Li, Z., Zhang, J., Zhuang, T., Zhuang, T., and Wang, Q., (2018). Metal Surface Defect Detection Based on MATLAB, in IEEE IAEAC '18 Proc., pp. 2365-2371.

Liu, J. and Li, Y., (2007). Algorithm for Image Correlation Matching. Geomat

Liu, W., (2020). Towards Visually Explaining VariationalAutoencoders. IEEE/CVF CVPR 20 Proc., pp. 8642-8651.

Lin, C.F. and Lin, S.F., Hwang, H.C., Tu, K.H., Chen, Y.C., and Weng J.C, (2019). Real-Time Imge-Based Defect Inspection System of Internal Thread for Nut. IEEE Transactions on Instrumentation and Measurement, vol. 68, no. 8, pp. 2830-2848.

Louis, B., (2022). Introduction to Convolutional Neural Networks (CNNs) | The Most Popular Deep Learning architecture | What I Artificial Intelligence | Medium. Available online: https://medium.com/what-is-artificial-intelligence/introduction-to-convolutional-neural-networks-cnns-the-most-popular-deep-learning-architecture-b938f62f133f

Martinez, P., Al-Hussein, M., and Ahmad, R., (2020). Intelligent Vision-Based Online Inspection System of Screw-Fastening Operations in Light-Gauge Steel Frame Manufacturing. Advanced Manufacturing Technology, vol. 109, no. 3-4, pp. 645-657.

Mishra, R., Reddy, G.Y.S. and Pathak, H., (2021). The Understanding of Deep Learning: A Comprehensive Review. Math. Probl. Eng. pp. 1-5.

Mohammad, M.T., (2023). Theoretical Understanding of Convolution Neural Network: Concepts, Architectures, Application, Future Directions.

Morbidelli, P., Carrera, D., Rossi, B., Fragneto, P., and Boracchi, G., (2020). Augmented Grad-CAM: Heat-Maps Super Resolution Through Augmentation. IEEE ICASSP ‘20 Proc., pp. 4067-4071.

Nwankpa, C., Ijomah, W., Gachagan, A. and Marshall, S., (2018). Activation Functions: Comparison of treands in Practice and Research for Deep Learning. arXiv:1811.03378.

Penatii, O.A.B., Nogueira, K., and Dos Santos, J.A., (2015). Do Deep Features Generalize from Everyday Objects to Remote Sensing and Aerial Scenes Domains. IEEE CVPRW 15 Proc., pp. 44-51.

Prabuwono, A.S., (2019). Automated Visual Inspection for Bottle Caps Using Fuzzy Logic.TEM Journal, vol. 8, no. 1, pp. 107-112.

Prakash, K.B., Kannan, R., Alexnder, S.A. and Kanagachidambaresan, G.R., (2021). Advance Deep Learning for Engineering and Scientists: A Practical Approach; Springer: Berlin/Heidelberg, Germany.

Priede, J., (2012). Implementation of Quality Management System ISO 9001 in the World and its Strategic Necessity. Procedia Social and Behavioral Sciences, vol. 58, pp. 1466-1475.

Qassim, H., Verma, A., and Feinzimer, D., (2018). Compressed Residual-VGG16 CNN Model for Big Data Places Image Recognition. IEEE CCWC ‘18 Proc., pp. 168-175.

Reddi, S.J., Kale, S., and Kumar, S., (2018). On the Convergence of Adam and Beyond, in ICLR 18 Proc., pp. 1-23.

Rezaei-Malek, Mohammadi, M., Dantan, J.Y., Siadat, A., and Tavakkoli-Moghaddam, R., (2019). A Review on Optimisation of Part Quality

Inspection Planning in a Multi-Stage Manufacturing System. International Journal of Production Research, vol. 57, no. 15-16, pp. 4880-4897.

Rodriguez, R., Gonzalez, C.I., Martinez, G.E. and Melin, P., (2021). An Improved Convolutional Neural Network Based on a Parameter Modification of the Convolution Layer. In Fuzzy Logic Hybrid Extensions of Neural and Optimization Algorithms: Theory and Applications; Springer: Cham, Switzerland, pp. 125-147.

SaiBalaji. (2020). Binary Image classifier CNN using TensorFlow. Techiepedia.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. and Chen, L.C., (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks, in IEEE/CVF CVPR '18 Proc., pp. 4510-4520.

Sarker, I.H., (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. SN Computer Science, vol. 2, pp. 1-21.

Scherer, D., Muller, A., and Behnke, S., (2010). Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition. ICANN '10 Proc., pp. 92-101.

Selvaraju, R.R., Cogswell, M., Das A., Vedantam, R., Parikh, D., and Batra, D., (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. IEEE ICCV '17 Proc., pp. 618-626.

Shrestha, A. and Mahmood, A., (2019). Review of deep learning algorithms and architectures. IEEE Access, vol. 7, pp. 53040-53065.

Simonyan, K., Vedaldi, A., and Zisserman, A., (2013). Deep Fisher Networks for Large Scale Image Classification. Advances in Neural Information Processing Systems, vol. 26, pp. 163-171.

Simonyan, K., and Zisserman, A., (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. ICWS ‘18 Proc., pp. 375-378.

Song, L., Li X., Yang, Y., Zhu, X., Guo, Q., and Yang H., (2018). Detection of Micro-Defects on Metal Screw Surfaces Based on Deep Convolutional Neural Networks. Sensors, vol. 18, no. 11, pp. 3709-3723.

Staar, B., Lutjen, M., and Freitag, M., (2019). Anomaly Detection with Convolutional Neural Networks for Industrial Surface Inspection. Procedia CIRP, vol. 79, pp. 484-487.

Suzuki, C., Be., (1985). Topological structural analysis of digitized binary images by border following. Comput. Vis. Graph. Image Process. 30,32-46.

Szegedy, C., Loffe, S., Vanhoucke, V., and Aleni, A., (2017). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. AAAI ‘17 Proc., pp. 4278-4284.

Tan, M. and Le, V, (2019). EfficientNet: Rethinking Model Scaling for Convolution Neural Networks. ICML ‘19 Proc., pp. 6105-6114.

Tao, F., Zuo Y., Xu, L., and Zhang, L., (2014). IoT-Based Intelligent Perception and Access of Manufacturing Resource Toward Cloud Manufacturing. IEEE Transactions on Industrial Informatics, vol. 10, no. 2, pp. 1547-1557.

Ukida, H., (2007). Visual Defect Inspection of Rotating Screw Heads, in SICE ‘07 Proc., pp. 1478-1483.

VomBrocke, J., Simons, A., Riemer, K., Niehaves, B., Plattfaut, R., and Cleven, A., (2015). Standing on the Shoulders of Giants: Challenges and

Recommendations of Literature Search in Information Systems Research. Communications of the Association for Information Systems, vol. 37, pp. 205-224.

VomBrocke, J., Simons, A., Riemer, K., Niehaves, B., Plattfaut, R., and Cleven, A., (2009). Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process. 17th European Conference on Information Systems Procedure, pp. 2206-2217.

Wang, G., (2018). Interactive Medical Image Segmentation Using Deep Learning with Image-Specific Fine Tuning. IEEE Transactions on Medical Imaging, vol. 37, no. 7, pp. 1562-1573.

Wang, J., Ma, Y., Zhang, L., Gao, R.X. and Wu, D., (2018). Deep Learning for Smart Manufacturing: Methods and Applications. Journal of Manufacturing Systems, vol. 48, pp. 144-156.

Wang, T., Chen, M., Qiao and Snoussi, H., (2018). A Fast and Robust Convolutional Neural Network-Based Defect Detection Model in Product Quality Control. International Journal of Advanced Manufacturing Technology, vol. 57, no. 15-16, pp. 4880-4897.

Weiss, K., Khoshgoftaar, M., and Wang, D., (2016). A Survey of Transfer Learning," J. Big Data, vol. 3, no. 1, pp. 1– 40.

Wen, L., Lixin, D., Dong, X., and Tsang, I.W., (2014). Learning with Augmented Features for Supervised and SemiSupervised Heterogeneous Domain Adaptation. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, no. 6, pp. 1134-1148.

Wu, Q., Han, M., and Zeng, X., (2019). "Single-Scale Workpiece Defect Detection Based on Deep Learning," 12th International Congress on Image and Signal Processing in CISP-BMEI '19 Proc., pp. 1-5.

Yadav, S., and Shukla, S., (2016). Analysis of K-fold Crossvalidation over Hold-out Validation on Colossal Datasets for Quality Classification. IEEE IACC ‘16 Proc., pp. 78–83.

Yang, J., Li, S., Wang, Z., and Yang, G., (2019). Real-Time Tiny Part Defect Detection System in Manufacturing Using Deep Learning. IEEE Access, vol. 7, pp. 89278–89291.

Yildiz, E. and Worgotter, F., (2019). DCNN-Based Screw Detection for Automated Disassembly Processes, in SITIS ‘19 Proc., pp. 187–192.

Zhai, W., Zhu, J., Cao Y., and Wang, Z., (2018). A Generative Adversarial Network Based Framework for Unsupervised Visual Surface Inspection, in IEEE ICASSP ‘18 Proc., pp. 1283–1287.

Zhang, B., Jaiswal, P., Rai, R., Guerrier, P., and Baggs, G., (2019). Convolutional Neural Network-Based Inspection of Metal Additive Manufacturing Parts. Rapid Prototyping Journal, vol. 25, no. 3, pp. 530-540.

Zhang, J., Huang, J., Chen, X. and Zhang, D., (2019). How to fully exploit the abilities of aerial image detectors. In Proceedings of the IEEE International Conference on Computer Vision Workshop, Seoul, Republic of Korea, pp. 27-28.

Zhang, Z., (2018). Improved Adam Optimizer for Deep Neural Networks. In Proceedings of the IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), Banff, AB, Canada, 4-6 June.

Zhou, F., Liu, G., Xu, F., and Deng, H., (2019). A Generic Automated Surface Defect Detection Based on a Bilinear Model. Applied Science, vol. 9, no. 15, pp. 1–17, Art. no. 3159.

Zhou, A., Zheng, H., Li M., and Shao, W., (2020). Defect Inspection Algorithm of Metal Surface Based on Machine Vision. ICMTMA ‘20 Proc., pp. 45–49.

Zoph, B., Vasudevan, V., Shlens, J., and LeQ, V., (2018). Learning Transferable Architectures for Scalable Image Recognition. IEEE/CVF CVPR ‘18 Proc., pp. 8697–8710.