

Machine Learning approach to thermite weld defects detection and classification



By:

Mohale Molefe

213538364

Supervisor:

Prof Jules-Raymond Tapamo

A thesis submitted in fulfilment of the academic requirements for the
degree of Masters of Science in Computer Engineering

in the

School of Engineering

College of Agriculture, Engineering and Science

University of KwaZulu Natal

Durban, South Africa

JUNE 2021

Preface

The research presented in this dissertation was conducted at the University of KwaZulu Natal under the supervision of Prof. Jules-Raymond Tapamo. I hereby declare that all the materials used in this dissertation are my own original work except where an acknowledgement is made in form of a reference. The work contained herein has not been submitted in part or whole for a degree at any other university.

Mohale Molefe
June 2021

Declaration 1: Supervisor

As the candidate's Supervisor, I agree to the submission of this dissertation.



Jules-Raymond Tapamo
June 2021

Declaration 2: Plagiarism

I, Mohale Molefe, declare that:

1. The research presented in this dissertation, except where otherwise stated is my original research.
2. The work presented in this dissertation has not been submitted to UKZN or another university for purposes of obtaining an academic qualification, whether by myself or any other.
3. This dissertation does not contain another person's data, pictures, graphs or other information unless specifically acknowledged as being sourced from other persons.
4. The research does not contain another person's writings unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:
 - (a) Their words have been re-written but general information attributed to them has been referenced.
 - (b) Where their exact words have been used, then their writings have been placed in italics and inside quotation marks and referenced.
5. This dissertation does not contain texts, graphics or tables copied and pasted from the internet, unless specifically acknowledged, and the source being detailed in the thesis and in the Reference section.

Mohale Molefe
June 2021

Declaration 3: Publications

I, Mohale Molefe, declare that the following publications came out of this dissertation.

1. M. E. Molefe and J. R. Tapamo, "Classification of Thermite Welding Defects using Local Binary Patterns and K Nearest Neighbors", 2021 Conference on Information Communications Technology and Society (ICTAS), pp. 91-96, 2021
2. M. E. Molefe and J. R. Tapamo, "Classification of Rail Welding Defects based on the Bag of Visual Words Approach", Lecture Notes in Computer Science, vol. 12275, pp. 255-272, 2021
3. M. E. Molefe, J. R. Tapamo, T. J. Mahlatji and S. S. Vilakazi, "Application of image processing and deep learning to automated rail weld defect classification", South African Heavy Haul Association (SAHHA2021), Kempton park, South Africa, [To appear in SAHHA2021 Conference]
4. M. E. Molefe and J. R. Tapamo, "Combining Multi-Layer Perceptron and Local Binary Patterns for Thermite Weld Defects Classification", Pan-African Artificial Intelligence and Smart Systems Conference (Submitted)
5. M. E. Molefe and J. R. Tapamo, "A Comparative Study of Image Processing and Machine learning Methods for Classification of Thermite Weld Defects", To be submitted to IEEE Transactions on Industrial Informatics

Mohale Molefe
June 2021

Acknowledgement

I would like to express my special thanks of gratitude and appreciation to my Supervisor, Prof. Jules Raymond Tapamo, for his support and guidance throughout this research. His friendly attitude and insightful feedbacks made it easy for me to work with him.

I am also thankful to my mentor and colleague Dr. Joshua Maumela whose knowledge and expertise was invaluable through this research.

I would also like to thank Thato Mahlatji, Refilwe Ndlela and Siboniso Vilakazi, who assisted with providing the dataset used to conduct experiments in this research work.

Lastly, I would like to acknowledge the Transnet Radiography specialists who categorised different thermite welding defects from the obtained dataset.

Abstract

The defects formed during the thermite welding process between two sections of rails require the welded joints to be inspected for quality purpose. The commonly used non-destructive method for inspection is Radiography testing. However, the detection and classification of various defects from the generated radiography images remains a costly, lengthy and subjective process as it is purely conducted manually by trained experts. It has been shown that most rail breaks occur due to a crack that initiated from the weld joint defect that was not detected. To meet the requirements of the modern technologies, the development of an automated detection and classification model is significantly demanded by the railway industry. This work presents a method based on image processing and machine learning techniques to automatically detect and classify welding defects. Radiography images are first enhanced using the Contrast Limited Adaptive Histogram Equalisation method; thereafter, the Chan-Vese Active Contour Model is applied to the enhanced images to segment and extract the weld joint as the Region of Interest from the image background. A comparative investigation between the Local Binary Patterns descriptor and the Bag of Visual Words approach with Speeded Up Robust Features descriptor was carried out for extracting features in the weld joint images. The effectiveness of the aforementioned feature extractors was evaluated using the Support Vector Machines, K-Nearest Neighbours and Naive Bayes classifiers. This study's experimental results showed that the Bag of Visual Words approach when used with the Support Vector Machines classifier, achieves the best overall classification accuracy of 94.66%. The proposed method can be expanded in other industries where Radiography testing is used as the inspection tool.

Contents

| | | |
|----------|--|-----------|
| 1 | General Introduction | 1 |
| 1.1 | Introduction | 1 |
| 1.2 | Problem Statement | 3 |
| 1.3 | Motivation | 4 |
| 1.4 | Main Aim and Specific Objectives | 5 |
| 1.5 | Study Limitations | 5 |
| 1.6 | Research Contributions | 5 |
| 1.7 | Dissertation Outline | 6 |
| 2 | Literature Review | 7 |
| 2.1 | Introduction | 7 |
| 2.2 | An Overview of Machine Learning | 7 |
| 2.3 | Application of Machine Learning in Railway | 9 |
| 2.3.1 | Shallow Learning-based Algorithms | 9 |
| 2.3.2 | Deep Learning-based Algorithms | 12 |
| 2.4 | Classification of Defects in Radiography Images | 15 |
| 2.4.1 | Image Enhancement | 15 |
| 2.4.2 | Image Segmentation and Weld Joint Extraction | 17 |
| 2.4.3 | Feature Extraction and Classification | 21 |
| 2.5 | Conclusion | 24 |
| 3 | Materials and Methods | 27 |
| 3.1 | Introduction | 27 |
| 3.2 | Image Enhancement | 28 |
| 3.2.1 | Histogram Equalisation | 28 |
| 3.2.2 | Contrast Limited Adaptive Histogram Equalisation | 29 |
| 3.2.3 | Algorithm for Image Enhancement using CLAHE | 29 |
| 3.3 | Region of Interest Extraction | 30 |
| 3.3.1 | Active Contour Models | 31 |
| 3.3.2 | Weld Joint Segmentation | 35 |
| 3.3.3 | Weld Joint Extraction | 35 |

| | | |
|----------|---|-----------|
| 3.3.4 | Post-Processing using Morphological Operations | 36 |
| 3.3.5 | Algorithm for Weld Joint Segmentation and ROI extraction | 37 |
| 3.4 | Feature Extraction | 37 |
| 3.4.1 | Feature Extraction using Local Binary Patterns | 39 |
| 3.4.2 | Feature Extraction using Speeded Up Robust Features | 42 |
| 3.4.3 | Image Representation using Bag of Visual Words | 47 |
| 3.5 | Feature Classification | 49 |
| 3.5.1 | Feature Classification using Support Vector Machines | 50 |
| 3.5.2 | Feature Classification using the K-Nearest Neighbors | 54 |
| 3.5.3 | Feature Classification using Naive Bayes | 55 |
| 3.6 | Evaluation Methods | 57 |
| 3.6.1 | Confusion Matrix | 57 |
| 3.6.2 | K Fold Cross-Validation | 58 |
| 3.7 | Conclusion | 59 |
| 4 | Experimental Results and Discussion | 61 |
| 4.1 | Introduction | 61 |
| 4.2 | Dataset Description | 61 |
| 4.3 | Image Enhancement | 62 |
| 4.3.1 | Parameter Evaluation | 63 |
| 4.4 | Weld Joint Extraction | 64 |
| 4.5 | Feature Classification | 67 |
| 4.5.1 | Classification of the Local Binary Patterns Features | 67 |
| 4.5.2 | Best Classifier for LBP Features | 74 |
| 4.5.3 | Classification of the Bag of Speeded Up Robust Features | 75 |
| 4.5.4 | Best Classifier for BoSURF Features | 80 |
| 4.6 | Best Method for Detection and Classification of Thermite Weld defects | 81 |
| 4.7 | Conclusion | 82 |
| 5 | Conclusion and Future Work | 83 |
| 5.1 | Dissertation Conclusion | 83 |
| 5.2 | Recommendation for Future work | 85 |
| | Bibliography | 87 |

List of Figures

| | | |
|------|--|----|
| 1.1 | Track structure | 1 |
| 1.2 | Thermite weld process and the weld joint formed | 2 |
| 1.3 | Some examples of thermite welding defects [1] | 3 |
| 1.4 | Train derailment site on the coal line [2] | 4 |
| 3.1 | System diagram of the proposed method | 27 |
| 3.2 | Image enhancement: (a) Original image and (b) Image enhanced using CLAHE. | 29 |
| 3.3 | Image segmentation: (a) Application of Chan-Vese ACM and (b) segmented image. | 35 |
| 3.4 | RoI extraction: (a) Image with bounding box and (b) Cropped image. | 35 |
| 3.5 | Post processing of the segmented image | 36 |
| 3.6 | Computation of the LBP code | 39 |
| 3.7 | Computation of a histogram vector using uniform LBP | 41 |
| 3.8 | Three box filter approximation of the second-order derivatives of Gaussian filters [3] | 43 |
| 3.9 | The scale, the filter sizes and octaves in SURF [4] | 44 |
| 3.10 | Non maximum suppression [5] | 44 |
| 3.11 | Keypoint orientation assignment [6] | 45 |
| 3.12 | Image representation using the bag of visual words | 47 |
| 3.13 | A hyperplane that separates the negative and positive samples [7] | 51 |
| 4.1 | Defect-less | 62 |
| 4.2 | Wormholes | 62 |
| 4.3 | Shrinkage cavities | 62 |
| 4.4 | Inclusions | 62 |
| 4.5 | Image Enhancement using CLAHE at varying clip factor values | 63 |
| 4.6 | Weld joint RoI steps | 64 |
| 4.7 | Weld joint extraction in a defect-less class | 65 |
| 4.8 | Weld joint extraction in Wormholes class | 65 |
| 4.9 | Weld joint extraction shrinkage cavities class | 66 |
| 4.10 | Weld joint extraction in Inclusions class | 66 |

4.11 Segmentation accuracy for each class 66

4.12 Classification accuracy of the K-NN classifier at varying LBP cell size parameter 69

4.13 Classification accuracy of the SVM classifier at varying LBP cell size parameter 72

4.14 Classification accuracy of the Naive Bayes classifier at varying LBP cell size parameter 74

4.15 Classification accuracy of the K-NN classifier at varying codebook size parameter 76

4.16 Classification accuracy of the SVM classifier at varying codebook size parameter 78

4.17 Classification accuracy of the Naive Bayes classifier at varying codebook size parameter 80

List of Tables

| | | |
|------|---|----|
| 2.1 | Recent publications on rail defect classification using machine learning techniques | 14 |
| 3.1 | LBP Parameters | 41 |
| 3.2 | Confusion matrix table | 57 |
| 4.1 | Confusion matrix using LBP and 5-NN at [6 14] cell size | 68 |
| 4.2 | Confusion matrix using LBP and 5-NN at [12 28] cell size | 68 |
| 4.3 | Confusion matrix using LBP and 1-NN at [30 70] cell size | 68 |
| 4.4 | Confusion matrix using LBP and 3-NN at [60 140] cell size | 68 |
| 4.5 | Confusion matrix using LBP and SVM ($\sigma = 4$) at [6 14] cell size | 70 |
| 4.6 | Confusion matrix using LBP and SVM ($\sigma=0.25$) at [12 28] cell size . . | 70 |
| 4.7 | Confusion matrix using LBP and SVM ($\sigma=0.5$) at [30 70] cell size . . . | 71 |
| 4.8 | Confusion matrix using LBP and SVM ($\sigma=0.5$) at [60 140] cell size . . | 71 |
| 4.9 | Confusion matrix using LBP and Naive Bayes at [6 14] cell size | 72 |
| 4.10 | Confusion matrix using LBP and Naive Bayes at [12 28] cell size | 73 |
| 4.11 | Confusion matrix using LBP and Naive Bayes at [30 70] cell size | 73 |
| 4.12 | Confusion matrix using LBP and Naive Bayes at [60 140] cell size . . . | 73 |
| 4.13 | Highest classification accuracy by each classifier for LBP features . . . | 74 |
| 4.14 | Confusion matrix using BoSURF and 3-NN at 200 codewords | 75 |
| 4.15 | Confusion matrix using BoSURF and 5-NN at 800 codewords | 75 |
| 4.16 | Confusion matrix using BoSURF and 5-NN at 1400 codewords | 76 |
| 4.17 | Confusion matrix using BoSURF and 3-NN at 2000 codewords | 76 |
| 4.18 | Confusion matrix using BoSURF and SVM ($\sigma = 0.5$) at 200 codewords | 77 |
| 4.19 | Confusion matrix using BoSURF and SVM ($\sigma = 4$) at 800 codewords . | 77 |
| 4.20 | Confusion matrix using BoSURF and SVM ($\sigma = 8$) at 1400 codewords . | 77 |
| 4.21 | Confusion matrix using BoSURF and SVM ($\sigma = 0.25$) at 2000 codewords | 78 |
| 4.22 | Confusion matrix using BoSURF and Naive Bayes at 200 codewords . . | 79 |
| 4.23 | Confusion matrix using BoSURF and Naive Bayes at 800 codewords . . | 79 |
| 4.24 | Confusion matrix using BoSURF and Naive Bayes at 1400 codewords . | 79 |
| 4.25 | Confusion matrix using BoSURF and Naive Bayes at 2000 codewords . | 79 |

4.26 Highest classification accuracy by each classifier for Bag of SURF (Bo-SURF) features 81

4.27 Highest classification accuracy achieved for LBP and BoSURF 81

List of Algorithms

| | | |
|---|--|----|
| 1 | Image enhancement using CLAHE | 30 |
| 2 | Image segmentation and weld joint (RoI) extraction | 37 |
| 3 | Feature extraction using LBP | 42 |
| 4 | Keypoint detection and description using SURF | 46 |
| 5 | Image representation using BoSURF approach | 49 |
| 6 | Feature classification using SVM | 53 |
| 7 | Feature classification using KNN | 55 |
| 8 | Feature classification using Naive Bayes | 57 |

Abbreviations

ACM Active Contour Models

AHE Adaptive Histogram Equalisation

BoSURF Bag of SURF

BoVW Bag of Visual Words

CDF Cumulative Distribution Function

CLAHE Contrast Limited Adaptive Histogram Equalisation

CNN Convolutional Neural Networks

CWR Continuously Welded Rails

DCNN Deep Convolutional Neural Networks

GHE Global Histogram Equalisation

GLCM Grey Level Co-occurrence Matrix

HoG Histogram of Oriented Gradients

K-NN K-Nearest Neighbours

LBP Local Binary Patterns

MLP Multi Layer Perceptron

NDT Non-Destructive Testing

RBF Radial Basis Function

RCF Rolling Contact Fatigue

RoI Region of Interest

RT Radiography Testing

SURF Speeded Up Robust Features

SVM Support Vector Machines

TFR Transnet Freight Rail

Chapter 1

General Introduction

1.1 Introduction

Railway transportation refers to passengers' transportation, various commodities, goods and services traded in as Cargo and Freight from one destination to another, using wheeled vehicles designed to run on rails. The South African railway industry is owned and managed by Transnet Freight Rail (TFR), which is one of six divisions of Transnet Ltd. TFR maintains an extensive rail network across South Africa that connects with other rail networks in Sub-Saharan regions and its rail network infrastructure represents approximately 80% of total infrastructure in Africa. The railway infrastructure is a complex and multi-purpose system that involves earthwork, tunnels, bridges, and a track structure. Each infrastructure system serves a specific purpose of assuring safe and reliable train transportation. Thus, proper maintenance planning and reliable infrastructure monitoring technologies are of great importance.



Figure 1.1: Track structure

Track structure is the most fundamental part of the railway infrastructure, and its primary purpose is to serve as a guideway for the train wheels and absorb dynamic stresses induced by the train motion. As illustrated in Figure 1.1, track structure comprises components such as rails, sleepers, ballast, and fasteners. The most critical and maintenance demanding component of the track structure is the rails. Unlike other components, rails are manufactured in sections and are joined together to form a continuous railway line during the installation process. The sections of the rails are usually welded together to form Continuously Welded Rails (CWR); the type of welding method used by TFR and other railway industries [8, 9] is thermite welding.

Also known as aluminothermic welding, thermite welding is a fast and inexpensive welding process used to join sections of rails permanently. Thermite welding is exothermic. It is characterised by the chemical reaction between iron oxide and aluminium to produce a superheated molten iron that fuses with the rails to form a weld joint. The advantages of thermite welding are its simplicity and mobility. Thus, it is mainly used on-site as a final step for the installation of rails, Figure 1.2 depicts the thermite welding process, and the weld joint produced. However, thermite welding is prone to the formation of defects on the welded joints. If these defects are undetected, they can lead to rail breaks and possible derailments, and loss of lives. Studies indicate that most rail breaks occur from cracks that initiate and propagate from the weld joints [2]. Therefore, it remains an important task to inspect the joints after the rails have been welded.



Figure 1.2: Thermite weld process and the weld joint formed

A wide range of Non-Destructive Testing (NDT) techniques has been used to inspect the weld joints for possible defects that could have occurred during the welding process. These include acoustic emission, eddy current, ultrasonic testing, and radiography testing. Radiography Testing (RT) is a commonly used NDT method across many railway industries [10]. RT has several advantages compared to other NDT methods as it allows radiography experts to examine and visualize defects from the generated images. The role of the radiography expert is to detect, classify, and accept

or reject the weld, based on the type of defect detected and the applicable radiography standards.

Five different types of welding defects can be produced from the RT methods. These are; lack of fusion, shrinkage cavities, inclusions, wormholes, and porosity. Lack of fusion defects occurs when there is an insufficient fusion between the weld joint and the parent rails during the welding operation. Shrinkage cavities are voids formed during the solidification of molten iron due to shrinkages; they are usually located in the upper web area of the rails. Porosity and wormholes are voids which are caused by gas entrapment. Porosity is spherical, and wormholes are elongated. Inclusions occur due to the presence of foreign materials; they are irregular in shape. Figure 1.3 illustrates some of the defects produced by the RT method.

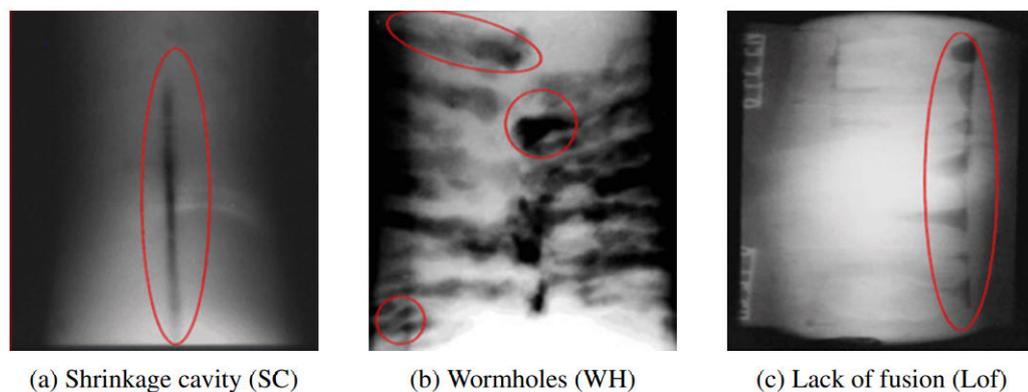


Figure 1.3: Some examples of thermite welding defects [1]

1.2 Problem Statement

The detection and classification of welding defects in the radiography images is conducted manually by a radiography expert with a load of experience. However, it remains a lengthy, costly, laborious, and subjective process even if conducted by the experts. In TFR, it usually takes up to two months to complete the detection and classification of defects from the time the weld joints are tested. This then leaves the weld joint exposed to a rail break, which could lead to a train derailment and, therefore, loss of lives and revenue. The reason behind a lengthy turnaround time is due to a lack of qualified experts to investigate thousands of weld joints tested monthly during the maintenance or installation of new rails.

One of many examples of a reported train derailment in TFR due to thermite weld defect was that of the coal line during the 2016/2017 financial year. Failure analysis

showed that the rail break occurred due to a crack that initiated at the weld joint. Figure 1.4 illustrates the derailment site on the coal line. Before the derailment, the coal line reported 32 rail breaks, 18 train cancellations, and 6520-minute delays. 58% of the rail breaks were on or adjacent to the weld joint [2]. These statistics indicate that the manual detection and classification of welding defects using human expertise is unacceptable. Thus, there is a need to develop an automatic defect detection and classification system that will address the shortcoming of the manual process.



Figure 1.4: Train derailment site on the coal line [2]

1.3 Motivation

Railway transportation plays a significant role in developing the South African economy, and industrial growth. Failures such as rail breaks due to thermite weld defects are directly linked to train derailments. At present, the detection and classification of thermite weld defects are performed manually by a trained radiography expert. However, manual inspection is problematic due to its low efficiency, lack of objectivity, high false alarm rate, and lengthy turnaround time. Additionally, the results are entirely dependent on the capability of the inspector to detect, classify, and assess the criticality of defects. Thus, there is a demand for the development of an efficient and accurate system that can detect and classify thermite weld defects automatically. The development of an automated system will assure that defects are detected and classified immediately after the weld joints are tested. This will significantly reduce the turnaround time and allow the maintenance teams to immediately remove the weld joints that possess a risk to rail breaks.

Over the years, computer vision technologies have been studied for various applications in the railway industry. Some of the successful applications of computer vision in the railway industry have been the development of the automatic classification of rail surface defects and railway fastener monitoring systems [11, 12, 13, 14]. However, the development of a computer vision-based system for the detection and

classification of thermite weld defects remains an area that has not been explored. This work is an attempt to automate the process by using image processing and machine learning techniques.

1.4 Main Aim and Specific Objectives

The main aim of this research work is to develop a computer vision-based model for the detection and classification of thermite weld defects. In an attempt to achieve the main aim, the following specific objectives have been set.

- Perform a literature review on existing techniques for the detection and classification of rail defects.
- Perform a literature review on existing techniques for the detection and classification of welding defects in radiography images.
- Develop an algorithm to improve the quality of the thermite weld images.
- Develop an algorithm to segment and extract the weld joint as the Region of Interest.
- Investigate several feature extraction methods and classification algorithms.
- Propose an efficient method for weld joint defect classification.

1.5 Study Limitations

Due to the scarcity of dataset, this research only consists of three types of thermite weld defects: wormholes, shrinkage cavities, and inclusions. Thus, only these three types of defects are classified.

1.6 Research Contributions

This work makes the following contributions to the railway and radiography industries.

- The development of a histogram equalisation based algorithm for improving the quality of thermite weld radiography images.
- The development of an Active Contour based algorithm for the extraction of the weld joint as the Region of Interest; Active contour methods are used to segment and further extract the weld joint as the Region of Interest from the background.

- The investigation of several feature extraction and classification algorithms; that has enabled the proposition of a suitable model for the weld joints defects classification.
- The design and implementation of an efficient model for thermite weld joint defect classification.

1.7 Dissertation Outline

- The present chapter has covered an introduction to rail thermite welding and processes, NDT method used to inspect the weld joint's quality and the type of weld defects produced using RT methods. Challenges associated with the manual classification of welding defects were stated, and the problem statement was defined.
- Chapter two presents the literature review on the current image processing and machine learning methods used to detect and classify defects in the railway and radiography industries. This includes methods based on deep learning and shallow learning algorithms.
- Chapter three discusses the material and methods for classification of thermite weld defects. It divides the methods into thermite weld image enhancement, weld joint extraction, feature extraction and feature classification.
- Chapter four provides the full experimental results and discussion obtained from various feature extraction and classification algorithms.
- Chapter five concludes the dissertation and provides the recommendation for future work.

Chapter 2

Literature Review

2.1 Introduction

Image processing and machine learning methods have enabled many railway practitioners to benefit from a wide variety of applications over the past years. The possibility of collecting data such as rail surface defects, missing fasteners, track geometry and welding defects has proven beneficial for efficient railway transportation and the development of predictive maintenance models. The use of machine learning methods in the railway industry started to gain popularity since the introduction of first-generation rail monitoring systems introduced by Trascino et al. [15]. These systems collected and stored various types of rail defect data, which later was reviewed by a trained personnel to make decisions. However, these systems did not incorporate automated detection and classification of rail defects in the captured data. As faster computing software became available, several researchers started introducing image processing and machine learning framework with high automation capabilities. Therefore, this chapter reviews some of the successful applications of machine learning in the railway and radiography industries. Section 2.2 presents an overview of machine learning. Section 2.3 discusses the recent applications of machine learning in the railway industry. Finally, Section 2.4 outlines some image processing and machine learning methods for extracting and classifying welding defects in radiography images. Section 2.5 concludes the chapter.

2.2 An Overview of Machine Learning

A machine learning algorithm is defined as an algorithm that can learn patterns from data. Three types of machine learning algorithms exist; supervised learning, unsupervised learning and reinforcement learning. Supervised learning algorithms learn from labelled training data that allows predictions about unseen or future data. Un-

supervised learning algorithms allow for the extraction of meaningful information from unlabeled data. For instance, the K-means clustering algorithm, one of the most popular unsupervised learning algorithms, takes unlabeled data in a high dimensional feature space and finds the target label by clustering features based on similarity measures [16]. Reinforcement learning is when the learning algorithm is no longer provided with the actual output values for its input values but with the score that tells the learning algorithm how good or bad its predictions are [17]. Machine learning algorithms can further be divided into shallow learning algorithms and deep learning algorithms.

The main difference between shallow and deep learning algorithms lies in their level of representation. Shallow algorithms use manually designed features and algorithms such as the Support Vector Machines (SVM) [18], K-Nearest Neighbours (K-NN) [19] and Random Forest [20] to train a shallow classifier. Deep learning algorithms, on the other hand, learn the features directly from raw data [21]. Shallow learning-based methods use manually designed or hand-crafted features to train shallow algorithms for learning the function that maps the predictive variables to the target variables. Additionally, these set of algorithms use a structured dataset in the form of a vector as inputs. For example, the SVM uses the feature vectors representing different samples in high dimensional feature space to learn a hyperplane that best separates between samples of two different classes. The unknown sample feature vector is assigned to a target function based on its relative position from the constructed hyperplane. A K-NN algorithm stores the feature vectors as part of the training process; the unknown sample is assigned a target function using the feature vectors nearest to the unknown sample.

Deep learning algorithms do not depend on hand-engineered or manually designed features, and they can learn features directly from the raw data (i.e. Raw images). Deep learning algorithms are often referred to as "representation learning", representation learning methods allow for machines to be fed with raw input data and automatically discovers the representation needed for a detection or classification task. For many applications, this property makes deep learning algorithms superior to shallow learning algorithms where features are extracted manually. Over the past couple of years, railway practitioners have also taken full advantage of deep learning for a wide variety of railway applications, including the detection, classification and monitoring of rail defects.

Convolutional Neural Networks (CNN) are a subset of deep learning algorithms widely used for a computer vision task. CNN with deep learning as defined in [21] are presentation-based learning methods containing multiple levels of presentations

obtained using non-linear convolutional layers to transform low-level representations (from the raw image) into more abstract levels. For example, extracting learned features from railroad image containing defects using CNN with three layers would have the first layer representing the presence or absence of the rail edges at specific locations in the image. The second layer would typically detect defects by spotting particular arrangements of edges, and lastly, the third layer may assemble edges combination that corresponds to a particular defect type.

The main difference between the standard neural network and CNN is that the convolutional layers replace the fully connected layers in the standard neural network. In contrast to fully connected layers, the neurons in convolutional layers are not connected to all the neurons in the previous and next layers; instead, the weights are shared between the group of layers. This sharing of weight is an important property of CNN as it allows the weight learned from the previous task to be applied in solving a new task; this is known as transfer learning.

2.3 Application of Machine Learning in Railway

In this section, the recent applications of machine learning algorithms in the railway industry are reviewed; Most specifically on the detection and classification of rail defects. The current literature divides these algorithms into shallow learning and deep learning.

2.3.1 Shallow Learning-based Algorithms

In shallow learning algorithms, researchers commonly relied on complex features extracted manually from images to train a shallow learning algorithm for detection and classification of defects in rails. Rajagopal et al. [22] used the Grey Level Co-occurrence Matrix (GLCM) to extract features in a dataset containing defective and defect-less rail images. The classification of the extracted GLCM features was carried out using the Neural Network classifier. The authors initially applied the image enhancement techniques to improve image quality before using the Gabor transform to obtain multiresolution images from the spatial domain.

A similar approach to the one presented in [22] was proposed by Yue et al. [23]. The authors combined geometric features with grey levels features to describe three surface defects, namely; scale peeling, crack stripping and tread crack. The multiclass classification of these defects was achieved using the AdaBoost classifier. According to the authors, the combination of the geometric features and grey levels features allows for detection of complex and random shape of the defect regions, which is an

improvement of the method proposed by Rajagopal et al. [22].

In [24], a method for detection and classification of images containing the scouring rail surface defects and defect-less images was proposed. Several feature extraction algorithms were used for experiments, these included; Principal Component Analysis (PCA), Kernel Principal Component Analysis (KPCA), Singular Value Decomposition (SVD) and Histogram Match (HM). The comparative analysis was achieved using the Random Forest as a classification algorithm. The experimental results conducted showed that the PCA gives higher classification accuracy while the HM achieved faster feature extraction and training time. This method was not effective as foreign objects on acquired images were detected as defects.

A vision-based method to automatically detect the absence or presence of fasteners in rails was proposed in [25]. Image pre-processing and feature extraction was done using the b-dimensional wavelet transform and PCA, respectively. While the Multi Layer Perceptron (MLP) classifier was used for recognising fasteners. This method lacked robustness against changes in illumination. Gilbert et al. [26] proposed a method based on the Histogram of Oriented Gradients (HoG) features and the linear SVM classifier to detect and classify defective fasteners on rail track images. The authors first applied image thresholding techniques to suppress noise and extract the fastener from the background. Though the use of HoG features allows for a reduction in intra-class variation, the issue of fastener image variations in viewpoints was not discussed.

A particular rail surface defect type called squats is usually caused by the Rolling Contact Fatigue (RCF). Gao et al. [27] made use of three different data sources consisting of ultrasonic, eddy current and rail surface images to detect squats more reliably. Features extracted from three data sources were grouped using a clustering algorithm and fed into the SVM classifier trained to detect squats. This method lacked accuracy, and the feature extraction process was slow. Jiang et al. [28] combined the laser ultrasonic technology and hybrid intelligent method to achieve fast classification and evaluation of RCF in different depths. The ultrasonic scanning systems collected data samples from different locations of the defects. Their method used Wavelet Packet Transform (WPT) to decompose the signal of the defects in different depths; KPCA to eliminate the redundancy of original feature set and the SVM classifier to classify defects in different depths.

Zhang et al. [29] proposed an automatic railway visual detection system that detects surface defects like squats, spalling and flaking. The authors used the vertical projection and grey contrast algorithms to extract the rail from the background im-

age. A curvature filter was also applied on the extracted rails to eliminate the noise and keep only the essential details. The detection of surface defects was achieved using the Gaussian Mixture Model (GMM) as a segmentation technique based on the Markov Random Field (RMF). This method allowed for the extraction of rail, even in challenging backgrounds. However, the main challenge is that an assumption made was that the pixels in an image are independently distributed, and the prior distribution of the GMM is independent on the spatial relationship between the pixels and their neighbours. Thus, the GMM was more susceptible to noise and illumination changes.

Grace and Rao [30] performed an analytical study of real-time rail surface defect prediction using three machine learning classification algorithms, namely; Neural Network, Decision Trees and Random Forest. The algorithms were trained and validated using the dataset collected by the Train Recording Car (TRC). The experimental results showed that the Decision Trees classifier outperforms the other classification algorithms to classify low-risk and high-risk surface defects. Even though the classification accuracy achieved is impressive, the downside of this study is on the defect detection stage, some defects could not be detected at high TRC speed; thus, improvements are needed.

A method for inspecting weld defects in welded rails was proposed by Nunez et al. [31] based on the Axle Box Acceleration (ABA) measurements data. ABA measures the vibrations induced by the wheel-rail interaction and indicates an irregularity in the rail-based on the measured wheel-rail interaction data. The authors used the Hilbert based approach to process, detect and assess the quality of the weld based on numerous registered dynamic responses in ABA. The obtained results were, however, dependent solely on the dynamic responses from ABA. Furthermore, the acceleration data used corresponded widely to the vibrations induced by the track irregularities.

An improved method to predict weld defects and classify the track conditions from the predicted results was proposed by Yao and Tao [32]. The authors extracted a wide range of features, including manufacturing technologies of welds, related materials, influential environment factors, and welding engineers' marks. These features were then used to train the machine learning classification algorithms including the SVM, Random Forest and Logistic Regression. However, their method does not detect and classify different rail welding defects; It only detect whether there is the presence of welding defects on rails and based on the predicted results the track is classified as safe or at risk.

2.3.2 Deep Learning-based Algorithms

Deep learning has seen many successes in the medical and science fields and it is currently the state-of-the-art paradigm in image recognition and speech recognition. One of the earliest attempts to apply deep learning algorithms for rail defect detection is presented in [33]. The authors designed a CNN model with two layers to classify defective and non-defective rails from stereo images, however, the small dataset was used for training, and thus the model was vulnerable to overfitting. James et al. [12] employed a multiphase deep learning-based technique to detect rail surface discontinuities on rail images; their approach first performed image segmentation techniques to remove the railroad from the background and then the linear binary classifier classifies the rail as defective or intact, the classification accuracy achieved was impressive.

Shang et al. [34] proposed a two-stage method for rail inspection using image segmentation and CNN. Their method was designed specifically for two objectives; to extract the rail surface from the background and classify the rail as defective or defect-less. The rail surface was extracted using the Canny edge operator to detect edges as the boundaries. Subsequently, the rail classification as defective or defect-less was achieved using CNN based on the inception-v3 pre-trained model. This method achieved great classification results; however, it was implemented for a binary classification task. Additionally, the Canny edge operator did not guarantee successful detection of edges in every image.

Roohi et al. [35] developed a Deep Convolutional Neural Networks (DCNN) framework to automatically detect and classify four classes of rail defects, namely, welding defects, light squats, moderate squats, and severe squats. The authors claimed that feature extraction using DCNN is more robust and accurate than the traditional feature extraction methods used on a large dataset. Their framework comprised three convolutional layers, three max-pooling layers, and three fully-connected layers. Subsequently, the hyperbolic tangent (Tanh) function and the rectified linear unit (ReLU) were used as activation functions. The classification accuracy achieved was impressive but could be better with hyperparameter tuning of parameters such as the learning rate and optimiser. Furthermore, their framework does not detect and classify different types of welding defects.

The method proposed by Jamshid et al. [36] detected the squats and predicted its growth based on the video images and ultrasonic measurement data. The ultrasonic measurement data was used to derive the general characteristics of the squats, and the video image data was used to analyse the growth of the visual length of defects. As an improvement to their previous method in [26] where the SVM classifier was

used to classify the fastener defects. Gibert et al. [11] trained a CNN pipeline based on five convolutional layers to classify the condition of the fastener as good, missing, or defective. To make their pipeline more robust against unusual situations, the authors used image augmentation and resampling techniques to add more "hard to classify" images to their training dataset.

Yanan et al. [37] developed a rail surface defect detection method using the YOLO-v3 deep learning network. Grey scale input images were initially divided into equal cells, and within each cell, the height, width and centre coordinates of the defects was calculated using the dimensional clustering method. The authors further used a logistic regression algorithm to calculate the bounding box score; meanwhile, the predictions of defect class that the bounding box contains were achieved using the binary cross-entropy loss function. However, the classification results were not impressive, and the learning rate was set to a high value. A high learning rate allows a model to learn faster at the cost of the sub-optimal solution.

Recurrent Neural Networks (RNN) are another example of deep learning algorithms commonly used for sequential and time-series tasks. Long Short-Term Memory (LSTM) networks are a particular case of RNN, and they can handle the vanishing gradient problem of the standard RNN. Xu et al. [38] developed an LSTM model to detect and classify defective and non-defective rail surface based on the ultrasonic measurement data. The pulse sequence from the ultrasonic data was interpreted as the sequential task in the LSTM architecture. The LSTM memory cell was used to establish the surface defect classification pipeline.

Song et al. [39] conducted a comparative study to detect and classify the severity of rail shelling defects. The dataset used to conduct the experiment included images of four levels of rail shelling defects ranging from low risk to high risk. The authors compared two pre-trained CNN models, namely the Residual Neural Networks (ResNet) and the VGG-16 network, as well as the approaches based on the manually extracted features, including the HoG descriptor with SVM classifier. The authors presented the results in terms of computation cost and classification accuracy. Their experimental results showed that the ResNet model takes less computational cost and achieves the highest overall classification accuracy. Table 2.1 illustrates an overview of the current publications of rail defect classification using machine learning techniques.

Table 2.1: Recent publications on rail defect classification using machine learning techniques

| Year | Author | Type of defects | Method | Algorithm | Acc(%) |
|------|-----------------------|-----------------|------------------|--------------------------------|--------|
| 2014 | Soukup et al. [33] | Surface defects | Deep learning | CNN | 55 |
| 2015 | Gilbert et al. [26] | Surface defects | Shallow learning | HoG + SVM | 92 |
| 2016 | Roohi et al. [35] | Surface defects | Deep learning | DCNN | 92 |
| 2016 | Santur et al. [24] | Surface defects | Shallow learning | PCA + Random Forest | 85 |
| 2018 | Rajagopal et al. [22] | Surface defects | shallow learning | GLCM + Neural Network | 94.9 |
| 2018 | Gao et al. [27] | Surface defects | Shallow learning | Clustering + SVM | 78 |
| 2018 | Grace and Rao [30] | Surface defects | Shallow learning | Decision Trees | 92 |
| 2018 | James at al. [12] | Surface defects | Deep learning | DCNN | 92 |
| 2018 | Shang et al. [34] | Surface defects | Deep learning | Inception-v3 pre-trained model | 91.2 |
| 2018 | Yanan et al. [37] | Surface defects | Deep learning | YOLO-v3 pre-trained model | 98 |
| 2019 | Yeu et al. [23] | Surface defects | Shallow learning | Gray features + AdaBoost | 81.1 |
| 2020 | Yao et al. [33] | Welding defects | Shallow learning | PCA + SVM | 92 |
| 2020 | Song et al. [39] | Surface defects | Deep learning | VGG-16 pre-trained model | 92 |

As shown in Table 2.1, most researchers have applied machine learning techniques specifically on the detection and classification of rail surface defects. Although the condition monitoring of weld joint based on machine learning has been studied by several researchers [34, 40, 41], a limited amount of research work can be found on the specific subject of the detection and classification of rail thermite weld defects using image processing and machine learning. Thus, for this research work, the following section will review some related work in other industries that use radiography to detect and classify welding defects in radiography images. Furthermore, Table 2.1 shows that deep learning has been the most extensively used technique for detecting and classifying rail defects. However, deep learning algorithms require an extensive amount of data for their implementation. Given that the dataset presented in this work is limited, the following section will review recent applications of image processing and shallow learning techniques for the detection and classification of welding defects in radiography images.

2.4 Classification of Defects in Radiography Images

The detection and classification of welding defects in radiography images give a somewhat more exciting and distinct challenge than rail surface defects images acquired through video cameras. Radiography images are commonly characterised by low contrast and poor image quality; thus, image enhancement techniques are the first step towards successfully detecting defects. The weld joint where welding defects are present needs to be further extracted from the image background to eliminate noise and reduce computation cost. Furthermore, the feature extraction of defects in the weld joint, as shown in the literature is usually done using the hand-engineered feature extraction methods. Finally, the extracted features are used as inputs to train the classification algorithms. This section discusses the recent applications of image processing and machine learning techniques to detect and classify welding defects in related industries that use radiography to inspect the weld joint.

2.4.1 Image Enhancement

Radiography images are characterised by the presence of noise and low contrast; thus, image pre-processing techniques are usually the first step towards a successful defect detection and classification pipeline. Generally, the image pre-processing stage consists of applying the image enhancement techniques to improve the visibility of defects and image filtering techniques to reduce noise. Several researchers have studied numerous image enhancement techniques for improving the quality of radiography images containing welding defects.

In the research work conducted by Roumen et al. [42] for automatic detection of defects in radiography images, the authors used two-dimensional adaptive filter and two-dimensional linear filter for noise suppression and correction of uneven background. Their method enhances the images at the less computational cost compared to other Fast Fourier Transform filters. Mohamad and Halim [43] applied a circular average filter and logarithmic intensity contrast to enhance radiography images containing inclusions and porosity defects. A comparative study by Maher et al. [44] compared the ideal, Butterworth and Gaussian high pass filters for noise removal and enhancement of radiography images consisting of the lack of penetration and porosity defects. Through experimentation, the authors proved that Gaussian high pass filter provides a smooth transition between various bands of pixels intensity values.

Radiography images usually have low contrast and, in most cases, improvement is achieved by manipulating the contrast of the image. The original radiography image has its grey level distribution highly skewed to the darker side of its histogram.

The contrast stretching approaches attempts to improve the contrast in an image by stretching the dynamic range of intensity values the image contains. The contrast stretching algorithm was proposed in [45] to make weld defects features stand out more clearly on images characterised by low contrast. The authors further applied the median filter on the stretched images to remove noise. The median filter allows for noise removal in images with less blurring effect.

Tokhy et al. [46] applied contrast stretching and normalisation algorithms to enhance radiography images containing welding defects. In this method, the images were first normalised using low and high threshold values, then the value closest to the maximum and minimum values was computed and finally, the contrast stretching algorithm was applied according to the determined range of contrast values. In the study proposed by Abidin et al. [47], three pre-processing techniques for image enhancement and noise removal were applied, these included noise removal by the median filter, image enhancement by contrast stretching and image sharpening with Laplacian filter.

The disadvantage of enhancement methods based on contrast stretching technique is that these methods are confined to a linear transform function for mapping input values to output values; thus, the results obtained are less impressive. Another disadvantage of contrast stretching techniques is that the overall appearance of an image is not considered. These techniques only transform a particular intensity value in the input image and produce a processed single intensity value in the output image. Histogram equalisation techniques enhance the global appearance of the image by manipulating its histogram depending on the application. Histogram equalisation improves the low contrast image by stretching its histogram distribution such that it is evenly distributed across the entire range of grey intensity values.

In the research work proposed by Zahran et al. [48], the authors suggested using the median filter and Weiner filter to remove noise prior to enhancing radiography images with Global Histogram Equalisation (GHE). However, enhancing images using GHE as outlined in [49] is not ideal since it assumes no illumination changes in foreground and background image objects. Additionally, for images where there is a change in illumination, the GHE mapping gives unwanted results such as over enhancement of intensity values with high probability values.

Adaptive Histogram Equalisation (AHE) is another essential histogram equalisation technique implemented to address the shortcoming of the GHE. AHE operates by dividing an image into equal cells and then computing histogram equalisation according to grey intensity values of every cell. The work introduced by Attia [50]

compared contrast stretching, GHE, and AHE for enhancement of radiography images. The authors presented the results in terms of peak signal to noise ratio and mean squared error. The experimental results proved that AHE outperforms the GHE while the contrast stretching achieved the worst results. However, Zhihong et al. [49] states that AHE techniques tend to over enhance local noise content since enhancement is carried out in local image regions.

Another variant of the adaptive histogram-based enhancement techniques is the Contrast Limited Adaptive Histogram Equalisation (CLAHE) technique. CLAHE technique prevents the over-amplification of noise in homogenous regions introduced by AHE; it does this by clipping the histogram in each cell to a desired contrast expansion and size. CLAHE has been the technique of choice for enhancing radiography images. Dang et al. [51] compared CLAHE with AHE, fuzzy, global and local enhancement algorithms in radiography images conforming to weld segmentation. The work proposed by Koonsanit et al. [52] is even more impressive; the authors combined log-normalisation and CLAHE to come up with what they called N-CLAHE algorithm for enhancement of radiography images. Normalising images, according to the authors allows CLAHE not to overexpose radiography images. Further application of histogram equalisation-based techniques for radiography image enhancement of weld defects has been applied in [53, 54].

2.4.2 Image Segmentation and Weld Joint Extraction

In many computer vision tasks, it is recommended to extract the Region of Interest (RoI) from the image background before performing feature extraction. Not only does the elimination of the background results in the removal of noise, but it also reduces the computational cost for further image analysis. In the detection and classification of welding defects in radiography images, the RoI is the weld joint where features containing defect types are contained. Image segmentation are techniques used to define the RoI for further image analysis. These techniques operate by partitioning an image into different sub-regions of background and foreground (RoI) objects. This section reviews some of the commonly used image segmentation techniques deemed effective for weld joint RoI extraction by researchers. The current literature divides these techniques into edge-based segmentation methods and region-based segmentation methods.

Edge based Segmentation

Also known as the discontinuity-based segmentation, edge-based segmentation techniques partitions objects in an image by identifying abrupt changes in its grey level intensity values. The output of this type of segmentation is the set of edges that are

connected around the RoI, and they define the weld joint. Most edge-based segmentation techniques rely on the computation of first and second-order image operators for edge identification. For instance, the work proposed by Carasco and Merry [55] relied on the Canny edge operator to detect edges and segment the weld joint image consisting of steel manufacturing welding defects. The segmented weld joint images were compared to an ideal binary image developed manually by experts. Mirzaei et al. [56] compared the Sobel, Canny and Gaussian filter for segmentation of weld joint on the welding images database. Although there was no significant difference in the results obtained, the Gaussian filter yielded better detection of edges.

According to [56], first-order derivative operators such as the Sobel edge detector are sensitive to noise and double edge formation. Thus, additional processing techniques are required for effective edge detection. As outlined in [57], the Canny edge operator achieves good signal to noise ratio compared to first-order derivative operators. Additionally, non-maxima suppression means the weak edges are eliminated, and thus the formation of double edges is minimised. However, the Canny edge operator requires much time to run due to complex computation. Another segmentation technique which relies on the detection of edges is the edge-based Active Contour Models (ACM).

Image segmentation using ACM is one of the successful and widely used technique in image processing. ACM provides an attractive method of segmenting images since they always produce sub-regions with continuous boundaries, contrary to the first and second-order derivatives, often producing discontinuous boundaries. The ACM is known as snake model formulated by Kass et al. [58]. It is a method of surrounding the RoI boundary in an image by a 'snake like' closed contour, the closed contour then dynamically adapts to the edges of the RoI in the image under the influence of internal forces, image forces and external constraint forces.

In their work for unsupervised welding defect classification based on Gaussian mixture models (GMM) and exact shape parameters, Nacerredine et al. [59] used snake ACM for two primary objectives; weld joint segmentation and defects segmentation. The authors outlined the advantages of snake ACM models in terms of segmenting objects with irregular shapes. Another unsupervised classification of welding defects based on GMM and shape parameters proposed by Zhang et al. [29] made use of snake ACM for weld joint segmentation. Image denoising techniques, including the curvature filter, were initially applied to original images to minimise noise. Despite being the widely used image segmentation method, the snake ACM is sensitive to the initial contour position and shape. For example, an initial contour should be positioned near the RoI to minimise the computation time. Another significant dis-

advantage of a snake model is its inability to change with topology.

Region based Segmentation

In contrast to the edge-based segmentation methods where edges are first identified, region-based segmentation takes the opposite approach, beginning from the inside of the ACM and the growing outward until weld joint boundaries are encountered. The region-based segmentation techniques are considered to be more advantageous than edge-based since they consider regions area rather than local properties such as gradients. Thresholding is one of the most straightforward and most fundamental region-based segmentation technique in image processing [60, 61, 62]. Thresholding relies on a fundamental fact that the dynamic range of pixel values between the RoI and the background is different. The output of this segmentation technique is a binary image with RoI represented by a white region and the background represented by a dark region.

Mouhmadi et al. [63] extracted the weld joint from the background images containing welding defects in steel pipes using global thresholding. The authors addressed the issue of low contrast and noise by applying image enhancement techniques to the acquired images. Several other applications of global thresholding for weld joint extraction in radiography images can be found in [64, 65]. However, the most common disadvantage of global thresholding methods for weld joint segmentation is that these methods assumes the acquired images only consist of a bimodal histogram. Therefore such methods are generally not ideal for images with a non-uniform background where there are variations in illumination.

Most researchers have focused their attention on the local thresholding techniques for weld joint segmentation. In local thresholding, an image is divided into sub-regions, and within each sub-region, a fixed value for separating the foreground and background is determined. The study conducted by Naceredinne et al. [59] compared the global thresholding to the local thresholding techniques for weld joint segmentation of radiography images. The results obtained from their study indicated that global thresholding yields good results on images with good contrast. For images with non-uniform background, local thresholding was recommended.

The notable disadvantage of thresholding is that the obtained binary image cannot be exploited immediately because of superfluous information that must be removed. Thus, the post-processing step is usually used after thresholding. The method proposed by Mahmoudi et al. [66] which is the improvement to their previous work in [62], relied on local thresholding for weld joint segmentation. Morphological operations were used as a post-processing technique to remove residual spots and to fill

holes from the thresholded images. In [67], the method based on multiscale morphology was applied to weld joint images segmented by the iterative Otsu threshold algorithm.

Region-based ACM is another essential segmentation techniques used in image processing. Region-based ACM is based on the level set theory, and they define energy functionals based on the region statistics rather than local image gradient. Most region-based segmentation techniques are based on the Mumford and Shah model which approximate the image to a pixel-wise smooth representation [68]. Chan and Vese developed a mean curvature flow-based level set implementation of the specific Mumford-Shah energy functional [69].

In this model, the mean intensity of the pixels inside and outside the curve approximates the image to a smooth representation. The Chan-Vese model is commonly known as the ACM without edges because it can detect objects whose boundaries are not represented by image gradients. Just like the edge-based ACM, this model minimises the energy until the desired boundaries are reached. However, the stopping term is not necessarily dependent on the gradient information. The main advantage of the Chan-Vese ACM is that contours can be split or merged together depending on the topology changes.

Gharsallah and Braiek [70] used a level set ACM to segment welding defects in images characterised by uneven illumination and low contrast. The authors exploited the saliency map as a feature representing image pixels embedded into a region energy minimisation function. The saliency map is said to be able to represent small defects even in images with low contrast. The results obtained by the authors indicated that level set ACM is more robust for segmenting images with challenging contrast and background as well as good performance compared to edge-based segmentation methods. Boutiche et al. [71] segmented weld joint and welding defects using the Chan-Vese ACM, curve evolution and binary level set methods. Their method aimed to extract defects in radiography images with uneven illumination and calculate the defect parameters for another application.

2.4.3 Feature Extraction and Classification

The feature extraction and classification steps follow the weld joint ROI step, respectively. Feature extraction aims to extract the meaningful information contained within the weld joint. The extracted information is usually represented as a histogram feature vector per every weld joint image. Feature classification involves training a machine learning classifier using the extracted features to distinguish between different feature vectors representing different welding defects. Two layers of feature extraction techniques exist; low-level feature extraction and mid-level feature extraction. Low-level image features are extracted from image pixels, and they aim to be invariant to a variety of image transformation. Mid-level image features, on the other hand, aim to combine a set of low-level features into a richer representation of intermediate complexity.

The Low-level visual feature extraction techniques extract visual properties from certain regions of the image via pixel-level operation. The extracted features are commonly referred to as global or local, according to the relative area of those regions. A global feature is computed by considering the entire image, and it reflects the global characteristics of the image. In contrast, a local feature is computed over a small region of the image. This section reviews the welding defects detection and classification methods based on global and local feature extraction techniques.

Defect Classification based on Global Features

The methods based on geometric features and texture features have been the most commonly used global feature extractors for extracting defect features in radiography images. The geometric features describe the shape, size, location, and intensity information of the welding defects, while texture features provide important visual information. Mekhalfa et al. [72] applied the SVM and MLP classifiers in four welding defect types, namely solid inclusions, porosity, lack of penetration, and crack. The authors first applied histogram equalisation techniques to improve the images' quality before extracting a set of geometric features derived from the geometrical defect parameters. In their study, the SVM provided higher accuracy and a faster computational time compared to MLP classifier. The work proposed by Valavanis and Kosmopoulos [73] made use of geometric and texture features for a multiclass welding defect classification pipeline. The authors compared the SVM, Neural networks and K-NN classifiers; the SVM classifier achieved the highest overall classification accuracy in this work.

A method for automatic detection of weld defects from radiography images based on the SVM classifier was proposed by Shao et al. [74]. Three types of global fea-

tures were extracted, including defect area, average grey scale difference and grey scale standard deviation. These extracted features were then used as inputs to a classifier to distinguish non-defective images from defective. The results showed that the proposed method could reduce the undetected rate and false alarm compared to traditional defect detection methods. The method proposed by Hassan and Awan [75] classified the welding defects using the geometric features and Artificial Neural network (ANN). The extracted geometric features include the defect area, major axis, minor axis, solidity and perimeter. The initial step involved enhancing the image contrast using the histogram equalisation before segmenting the weld defect region using global thresholding and morphological operations.

Silva et al. [76] extracted four geometric features, including position, aspect ratio, and the roundness of various types of welding defects. Classification of the extracted features was achieved using the ANN classifier. Their method proved that the quality of the extracted features is more important than the quantity of the features. Hernandez et al. [77] extracted features describing the defect size, defect shape, defect location, and information intensity. These features are similar to the features extracted in [75]. The defects classified included the inclusions, porosity, longitudinal crack. The classification of the extracted features was achieved using the adaptive network-based fuzzy interference system (ANFIS). A similar method based on geometric features and ANFIS to classify five types welding defects was proposed in [78].

Global features are attractive because they produce a very compact representation of images where each image corresponds to a single point in a high dimensional feature space. Furthermore, global features require less computational cost compared to the requirements of local features. However, global features are not invariant to significant image transformations and are sensitive to clutter and occlusion. As a result, it is either assumed that an image contains only a single object or that good segmentation of the object from the background is available. The approach to overcoming these limitations, as stated by Ibrahim et al. [79] is to segment images into several regions with each region representing a single object. However, image segmentation is a challenging task that requires a high level understanding of the image content.

Defect Classification based on Local Features

Global feature's limitations are overcome by a local feature that finds interesting characteristics of the image content despite significant changes in illumination, occlusion, viewpoint and clutter, and the image does not need to be segmented. A local feature is computed over a relatively small region of the image. It is defined as a pattern in an image that differs from its immediate neighbourhood. A local feature in an image content can be points, edges, corners or small image patches [67]. Two

types of local descriptors are found in the literature, keypoint based descriptors and grid sampling-based descriptors. Keypoints are points such as corners and blobs, and their shape, scale and position are found using a feature detector. On the other hand, Grid sampling descriptors consist of patches of fixed size and shape placed on a regular grid across an image. This section reviews the classification of welding defects conducted by several researchers based on local feature extraction techniques.

The HoG descriptor is a grid sampling-based descriptor which has been very successful in facial recognition tasks [80, 81, 82]; it is invariant to illumination changes. Gao et al. [83] proposed a method for automatic detection and classification of welding defects in heating panels. The authors used the HoG descriptor for feature extraction while the kernel-based SVM was used as a classifier. However, their method was not suitable for rotated images. Liu et al. [84] proposed a rather more interesting weld defect classification pipeline based on CNN with three fully connected layers. Features in the second fully connected layer were extracted using the HoG descriptor. The authors then used ensemble methods to classify features in the last fully connected layer of the CNN. The pipeline gave good accuracy at the expense of large data requirements.

Local Binary Patterns (LBP) descriptor is one of the widely used grid sampling-based descriptors for extracting local features in images. It is invariant to illumination changes and rotation. The work proposed by Moghaddam et al. [85] compared the linear SVM and the K-NN classifiers to classify weld defects features extracted using the LBP descriptor. The authors considered three types of welding defects: lack of penetration, lack of fusion and external undercut. The first step was to improve the contrast of images using the two-dimensional filter before performing feature extraction. The K-NN outperformed the linear SVM classifier by a significant margin in terms of classification accuracy.

Mery et al. [86] conducted an empirical study to automatically detect weld defects in a large dataset of automotive radiography images. The authors compared 24 computer vision techniques, including deep learning, sparse representation and local descriptors. The experiments conducted by the authors showed that the best performance is achieved by the combination of the LBP descriptor and a linear SVM classifier. Moreover, the application of the LBP descriptor and the SVM classifier to detect and classify weld defects in radiography images can be found in [87, 88].

Feature extraction using the keypoint based descriptors include two main steps, keypoint detection and keypoint description. Keypoint detection aims to find interesting information or keypoints on the image that are invariant to a wide variety of im-

age transformations, and keypoint description aims to compute a descriptor vector that describes the local content of a keypoint. The Scale Invariant Feature Transform (SIFT) and the Speeded Up Robust Features (SURF) are the commonly used keypoint based descriptors. These descriptors aim to transform an image into vectors that are invariant to image transformations such as illumination changes, rotation, viewpoint, scale and affine transformation.

The method based on the SIFT descriptor and the SVM classifier was proposed in [5] to detect and classify steel defects, due to multiple features generated per image, the authors suggested feature reduction by the voting strategy based on the one versus all multiple classifiers. The SIFT descriptor yields a 64-dimensional feature vector describing the local content for every detected keypoint. Keypoints in SIFT are detected using the Difference of Gaussian (DoG) generated from image pyramid. On the other hand, the SURF descriptor was implemented to improve SIFT in terms of reduced feature vector length and faster detection of keypoints [89]. Kalai et al. [6] detected and classified the slag inclusions, porosity, lack of fusion and incomplete penetration defects on steel welding images. Features of these defects were extracted using the SURF descriptor while the Auto-Encoder Classifier (AEC) was employed for classification purposes. AEC was analysed for weld image classification using a different number of neurons in different hidden layers.

Despite achieving great results in many applications and being robust and invariant to many image transformations, keypoint based descriptors such as SIFT and the SURF consist of many feature vectors representing a single image. This yields a high dimensional feature space; thus, the computational cost is high, and the classification results are affected by outliers. This is because keypoint vectors could be classified as belonging to a different class label even though they came from the same image.

2.5 Conclusion

In Section 2.2, an overview of machine learning techniques was introduced, and it was noted that machine learning techniques could be divided into supervised learning, unsupervised learning and reinforcement learning. Supervised learning learns from the labelled dataset, while unsupervised learning finds interesting characteristics in the unlabelled dataset. Reinforcement learning is when the learning algorithms are provided with the score that tells the algorithm how good or bad its predictions are. It was further noted that machine learning algorithms could be divided into shallow learning algorithms and deep learning algorithms. Deep learning algorithms learn the features directly from raw data, while shallow learning algorithms rely on manually extracted features.

In Section 2.3, the recent applications of machine learning algorithms for the detection and classification of rail defects were investigated. The applications included methods based on shallow learning algorithms and deep learning algorithms, and the summary of the results obtained were presented in Table 2.1. It was observed that both shallow and deep learning algorithms had been used widely for the detection and classification of rail surface defects and very little work was found on the rail thermite welding defects. Thus, a further investigation of some related work in other industries that use radiography to detect and classify welding defects was conducted in Section 2.4. Given that the dataset presented in this work is limited, section 2.3 only investigated the methods based on image processing and shallow learning techniques.

In Section 2.4, the detection and classification of welding defects in radiography images was presented in terms of four steps, namely: Image enhancement, weld joint extraction, feature extraction and feature classification. The Image enhancement step aimed at improving the quality of the images for further image analysis. It was found that the commonly used image enhancement techniques are based on contrast stretching, histogram equalisation and median filter for noise removal after image enhancement. Additionally, it was found that histogram equalisation, most specifically the adaptive techniques, provides better enhancement of radiography images than contrast stretching techniques. Thus, the image enhancement technique to improve the quality of the images acquired for this work will be based on adaptive histogram equalisation techniques.

The techniques investigated for weld joint extraction included edge-based segmentation techniques and region-based segmentation techniques. It was observed that edge-based segmentation methods require the computation of edges in images using the first and second-order derivative operators. However, this could be a challenge for cases where the detection of edges in images is not feasible. The region-based segmentation on the other hand like the thresholding technique only uses the statistical information of background and foreground objects for segmentation. Furthermore, thresholding was found not suitable for segmenting images with uneven background. Another edge-based segmentation investigated was the Chan-Vese ACM; this method was found to have a better performance compared to edge-based segmentation methods since it can segment images with tribological changes.

Two feature extraction methods were investigated; these were the global feature extraction methods and local feature extraction methods. In this work, the feature extraction method must at least satisfy the following image transformation require-

ments: illumination invariance and rotation invariance. Global features investigated in the literature don't meet the above requirements. Two local feature extraction methods were reviewed, Grid sampling-based methods and keypoint based methods. The HoG descriptor only meets one of the requirements while the LBP descriptor meets both of the requirements; thus, the LBP descriptor will be used for experiments.

Two keypoint based descriptors were also investigated; SIFT and SURF descriptors. Both of these descriptors meet the requirements and they are also invariant to scale, however, the major disadvantage as found in the literature is that they both represent an image by many feature vectors and this is not ideal for training a classifier. Thus, in this work, a novel mid level image representation method that aims to combine the keypoint based features into a global image representation is proposed. The method is based on the Bag of Visual Words (BoVW) mid level image representation approach. The SURF descriptor will be used for this purpose since it has a small dimensional feature vector and it detects keypoints much faster than the SIFT descriptor. For feature classification, the SVM, K-NN and Naive Bayes classification algorithms will be used for experiments.

Chapter 3

Materials and Methods

3.1 Introduction

This chapter's main objective is to provide a mathematical pipeline of the methods used in this work to detect and classify thermite weld defects. As depicted in Figure 3.1, the methods are divided into thermite weld image enhancement, weld joint Region of Interest (RoI) extraction, feature extraction and feature classification. Images are initially enhanced to improve their quality; thereafter, the weld joint is extracted from the background of the enhanced images. Feature extraction is performed on the weld joint to extract defect features. The extracted features are then used to train and validate a classification algorithm. This chapter is structured as follows. First, image enhancement and weld joint RoI methods are presented in Sections 3.2 and 3.3, respectively. Then the feature extraction methods to extract defect features are outlined in Section 3.4. After that, the feature classification methods to classify the considered defects are discussed in Section 3.5, and finally, Section 3.6 discusses the evaluation methods. Section 3.7 concludes the chapter.

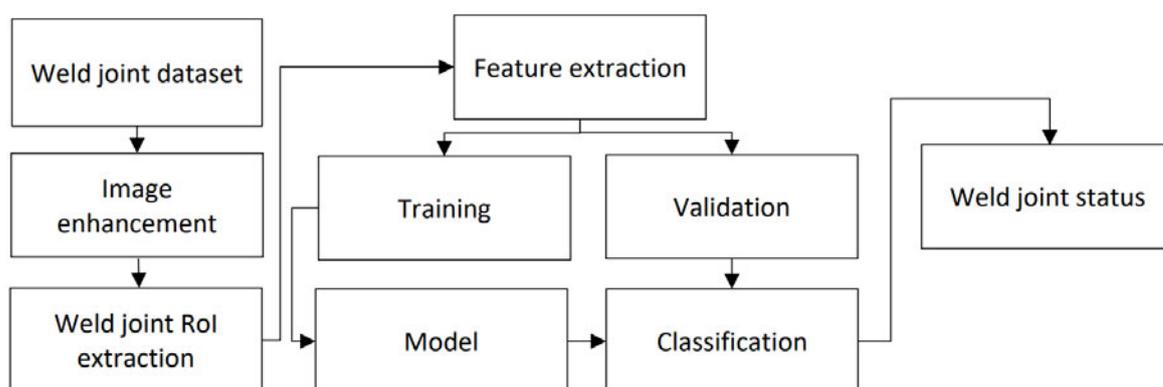


Figure 3.1: System diagram of the proposed method

3.2 Image Enhancement

Low pixel dynamic range intensity values characterise the collected thermite weld images, where pixels are either skewed to the right or to the left of the histogram. Thus, image enhancement techniques are required to improve the image quality such that the dynamic range of pixels is evenly distributed across the entire histogram. As discussed in Chapter 2, several image enhancement techniques have been used in the literature to enhance radiography images. These are divided into contrast stretching and Histogram equalisation. Contrast stretching enhances the quality of an image by increasing the dynamic range of the pixels. It takes the narrow range of intensity values in the normalised input image and produces a wide range of intensity values in the processed image. The disadvantage of the contrast stretching technique is that it is only confined to a linear transform function for mapping input values to output values. Furthermore, it is based on point processing, and it does not consider the overall appearance of the image [47]. Histogram equalisation offers more advantages than contrast stretching since the global appearance of the image can be enhanced by manipulating its histogram. Therefore, histogram equalisation is used in this work to determine a function that transforms an original image into an enhanced image. Histogram equalisation techniques can be divided into global-based and adaptive-based approaches.

3.2.1 Histogram Equalisation

Global Histogram Equalisation (GHE) is the basic histogram equalisation technique and it enhances an image by computing a global transform function using the histogram of the entire image. Given a thermite weld radiography image $I = (I(x, y))$ where $0 \leq x \leq N - 1$, $0 \leq y \leq M - 1$ with L discrete grey level intensity values denoted as $\{g_0, g_1, \dots, g_{L-1}\}$ and $I(x, y) \in \{g_0, g_1, \dots, g_{L-1}\}$ is the image intensity at spatial location (x, y) . Then, the normalised histogram or probability density function $p(g_k)$ of I is defined as:

$$p(g_k) = \frac{n_k}{n} \quad (3.1)$$

Where n_k is the total number of pixels in I with grey level g_k for $k = 0, 1, \dots, L - 1$ and $n = N \times M$ is the total number of pixels in I ; $p(g_k)$ is the Probability Density Function (PDF), and the plot of the PDF is known as the Histogram. From the obtained PDF of I , the Cumulative Distribution Function (CDF) is defined as:

$$cdf(g_k) = \sum_{j=0}^k P(g_j) = \sum_{j=0}^k \frac{n_j}{n} \quad (3.2)$$

GHE then maps the histogram of I into its entire range of grey values $\{g_0, g_{L-1}\}$ by using the CDF as the transform function. The transform function $T(g_k)$ is defined using the function $cdf(g_k)$ as:

$$T(g_k) = g_0 + (g_{L-1} - g_0)cdf(g_k) \quad (3.3)$$

Then the output image of the GHE is denoted by $G_I = (G_I(x, y))$, where $G_I(x, y)$ is expressed as:

$$G_I(x, y) = T(I(x, y)) \quad (3.4)$$

3.2.2 Contrast Limited Adaptive Histogram Equalisation

GHE does not perform well on images that contain local regions of low contrast. In such cases, the Adaptive Histogram Equalisation (AHE) technique is used. It divides an image into regions, and for each region, it calculates contrast enhancement based on the regions CDF's [90]. However, AHE tends to over enhance noise contents since it operates in local areas. Thus, the enhancement technique presented in this work is based on the Contrast Limited Adaptive Histogram Equalisation (CLAHE). CLAHE overcomes the noise enhancement artefact of AHE by clipping the histogram before using the CDF as a transform function. The clipping limit is calculated as [91]:

$$C.L = \frac{p \times q}{L} \left(1 + \frac{\rho}{100} (\delta_{max} - 1) \right) \quad (3.5)$$

Where $C.L$ is the clipping limit, $p \times q$ is the cell size of each region, ρ is the clip factor, and δ_{max} is the maximum allowable slope. Figure 3.2 shows the application of the CLAHE technique on the thermite weld image.

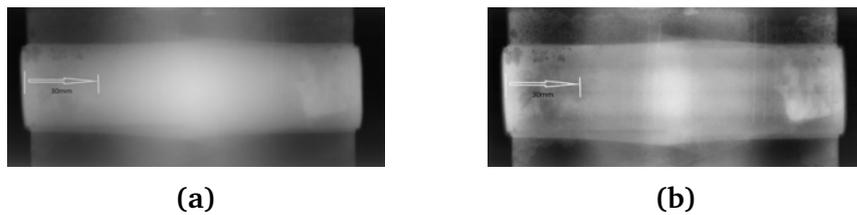


Figure 3.2: Image enhancement: (a) Original image and (b) Image enhanced using CLAHE.

3.2.3 Algorithm for Image Enhancement using CLAHE

The steps for performing image enhancement on the thermite weld images using the CLAHE technique are given in Algorithm 1.

Algorithm 1 Image enhancement using CLAHE

Require: Thermite weld images**Output:** Enhanced images

```
1: for each image  $I$  in the dataset do
2:   Divide into equal non-overlapping cells.
3:   for each cell do
4:     Calculate the histogram using Equation 3.1.
5:     Calculate the clip limit using Equation 3.5.
6:     Calculate the CDF using Equation 3.2.
7:     Calculate the transform function using Equation 3.3.
8:     Obtain the enhanced region using Equation 3.4.
9:   end for
10: end for
```

3.3 Region of Interest Extraction

The main focus of this section is to determine and extract the Region of Interest (RoI) from the enhanced thermite weld images. The RoI is the weld joint, and it must be removed from the background. Before extracting the weld joint, images must first be segmented to identify the region and the coordinates of the weld joint from the image background accurately. Additionally, the requirements of the segmentation technique in this work is to be able to segment the irregularly shaped weld joint RoI from the thermite weld images characterised by a complex background. As discussed in Chapter 2, image segmentation techniques such as Thresholding, Hough transform and Active Contour Models ACM are available for segmenting radiography images.

Image segmentation using Thresholding techniques is simple and fast to compute; however, Thresholding methods are effective for segmenting images with bimodal histogram. Furthermore, Thresholding techniques use a fixed threshold value to distinguish the foreground region from the background region; thus, their application is not effective for images characterised by a complex background. Another commonly used segmentation method is the Hough transform, the Hough transform allows for segmenting image regions consisting of various known shapes such as circles and ellipses; however, when the region to segment is irregular in shape, Hough transform is not effective [92].

Image segmentation using ACM is one of the successful and widely used segmentation techniques for a variety of tasks in image processing [93, 94, 95]. ACM provides an efficient way of using an energy function to drive the contour towards the

object's boundaries to segment, thus allowing irregularly shaped image regions to be segmented. Therefore, ACMs are used in this work to segment the weld joint; most specifically, the region-based ACM, as their advantages compared to other ACM techniques will be explained. This section is presented as follows. First, the mathematical approach of ACM for image segmentation is presented in Subsection 3.3.1. Then, the segmentation and extraction of the weld joint from image background is given in Subsections 3.3.2 and 3.3.3, respectively. Subsequently, the post-processing methods for removing residual spots in segmented images is presented in Subsection 3.3.4. Finally, the algorithm for extracting the weld joint RoI is outlined in Subsection 3.3.5

3.3.1 Active Contour Models

Image segmentation using ACM relates the segmentation problem to an optimisation problem, formulated in terms of the energy function that is constrained such that its minimum is achieved in correspondence to a contour that approximates the actual object boundaries. The optimisation is performed iteratively, starting from the initial contour position, then evolving towards regions that provide a better approximation of the object boundaries. Image segmentation using ACM can be divided into two categories; edge-based ACM and region-based ACM. Edge-based ACM relies on the detection of edges to stop the evolution of the contour on the actual object boundaries. In contrast, region-based ACM uses statistical information of the regions to be segmented. Two mathematical approaches for image segmentation using ACM exist; parameterised approaches and level set-based approaches.

Parameterised Approaches

A representation of the parameterised ACM is the snake model proposed by Kass et al. [58]. Snake model segment the image by first defining a snake-like contour around the RoI; then, under the influence of the energy function, the contour is driven towards the object and eventually stops at the boundary. Given the location of the parametric contour C in an image as: $C(s) = (x(s), y(s))$, then, the energy function which must be optimised is defined as:

$$E_{Snake} = \int_0^1 [E_{Int}(C(s)) + E_{Ext}(C(s))] ds \quad (3.6)$$

Where E_{int} represents the internal energy of the contour and E_{ext} represents the external forces. Internal energy encourages the contour to conform to a known shape preference; it serves to impose a piecewise smoothness constraints [96]. The internal energy at some point $C(s)$ on the curve is defined as:

$$E_{int} = (\alpha(s)|C'(s)|^2 + \beta(s)|C''(s)|^2)/2 \quad (3.7)$$

Where $C'(s)$ is the first-order derivative, and it makes the contour to act as a membrane (elasticity), and $C''(s)$ is the second-order derivative which allows the contour to act as a thin plate (rigidity). $\alpha(s)$ and $\beta(s)$ are the user-defined parameters which controls the relative importance of $C'(s)$ and $C''(s)$ respectively.

External forces attract the contour towards image features such as edges, lines and texture. They can be interpreted as a gravitational pull towards edges in an image. At a contour location $C(s)$ in image I , the external force is calculated as:

$$E_{Ext} = \int_0^1 -\|\nabla I(C(s))\|^2 \quad (3.8)$$

The snake model provides an attractive method of segmenting images since it produces sub-regions with continuous boundaries, contrary to the first and second-order derivatives, which often produces discontinuous boundaries [97]. Despite being the widely used image segmentation method [98, 99, 100], The snake model is said to be sensitive to the initial contour position and shape. For example, an initial contour should be positioned near the RoI to minimise the computational cost. Another significant disadvantage of a snake model is its inability to change with topology [101].

Level Sets Methods

Level sets ACM were first introduced by Osher and Sethian [102]. The difference between the parametric ACM and the level sets ACM is that the latter implements the contour via a variational level set method. The contour is represented implicitly by a function $\phi(x, y)$ called a level set function, where (x, y) is the pixel location in the image domain Ω . The contour C is defined as those pixels in Ω where the level set function is zero, and this is expressed as:

$$C = \{(x, y) \in \Omega : \phi(x, y) = 0\} \quad (3.9)$$

The level set function can be interpreted as the distance function with respect to the contour C . It is positive outside the contour, zero at the contour location and negative inside the contour. Given that the contour C moves with speed F in the normal direction, then the level set function $\phi(x, y)$ must satisfy the following level set equation:

$$\frac{\partial \phi(x, y)}{\partial t} = F|\nabla \phi(x, y)| \quad (3.10)$$

Two alternative approaches for level set segmentation exist; Geodesic ACM and Chan-Vese ACM. In Geodesic ACM, the gradient descent equation providing speed F is derived in terms of the contour and then implemented using the level set equa-

tion. This was done to derive the level set equation for a snake model. The energy function which must be minimised is defined as:

$$E(C) = \int g(C)dC \quad (3.11)$$

The above equation is minimum at the edges of the object and g is an edge indicator function defined as:

$$g(I(x, y)) = \frac{1}{1 + |\nabla I_\sigma(x, y)|} \quad (3.12)$$

Where $I_\sigma(x, y)$ is the smoothed image representing the spatial scale where the gradient is computed. The gradient descent equation providing the speed of the contour in the normal direction is given as:

$$\frac{dC}{dt} = g\kappa n + (n \times \nabla g)n \quad (3.13)$$

Where κ is the local curvature of C and n is the outer normal. Implementing Equation 3.13 in terms of the level set Equation 3.10 gives the level set equation for Geodesic ACM defined as:

$$\frac{\partial \phi(x, y)}{\partial t} = g(I(x, y))|\nabla \phi(x, y)|\text{div} \left(\frac{\nabla \phi(x, y)}{|\nabla \phi(x, y)|} \right) + \nabla g(I(x, y))\nabla \phi(x, y) \quad (3.14)$$

Geodesic ACM, as already mentioned, offers more advantages compared to a snake model as it allows the segmentation of images with changes in topology. However, it should be mentioned that Geodesic ACM relies on the edge detector to stop the evolution of the contour; this then makes segmentation a challenging task if detecting edges in an image is impractical or if image edges are smooth.

Another segmentation method based on the level set approach is the Chan-Vese ACM, proposed by Chan and Vese [69] for the Mumford-Shah [103] segmentation techniques. The Mumford-Shah model can detect contours without relying on gradient information. For instance, objects with very smooth edges or non-connected edges can be segmented. Given an image with two regions Ω_1 and Ω_2 representing the foreground and background objects, respectively; the heaviside step function is defined as:

$$H(\phi(x, y)) = \begin{cases} 1, & \text{if } \phi(x, y) \geq 0 \ ((x, y) \in \Omega_1) \\ 0, & \text{else } ((x, y) \in \Omega_2) \end{cases} \quad (3.15)$$

The Chan-Vese ACM is based on the Mumford-Shah model, and it segments the image by using the grey scale intensity information within regions as opposed to using the edge information. The Mumford-Shah energy function E_{cv} which must be minimised

for two energy regions E_1 and E_2 in Ω_1 and Ω_2 , respectively is expressed as:

$$E_{cv} = \int_{\Omega_1} |(I(x, y) - h_1)|^2 dx dy + \int_{\Omega_2} |(I(x, y) - h_2)|^2 dx dy + v|\partial\Omega_1| \quad (3.16)$$

Where h_1 and h_2 are the mean intensity values inside and outside the contour respectively, these are updated for each iteration, and $v|\partial\Omega_1|$ is the length of the boundary which is used as a regularising term. The Chan-Vese ACM in terms of the level set function ϕ can be written as:

$$E(h_1, h_2, \phi) = \int_{\Omega} ((I(x, y) - h_1)^2 - (I(x, y) - h_2)^2) H\phi(x, y) + \int_{\Omega} (I(x, y) - h_2)^2 dx dy + v \int_{\Omega} |\nabla H\phi(x, y)| dx dy \quad (3.17)$$

The mean intensity values h_1 and h_2 inside and outside the evolving contour, respectively are defined as:

$$h_1(\phi) = \frac{\int_{\Omega} I(x, y) H(\phi(x, y)) dx dy}{\int_{\Omega} H(\phi(x, y)) dx dy} \quad (3.18)$$

$$h_2(\phi) = \frac{\int_{\Omega} I(x, y) (1 - H(\phi(x, y))) dx dy}{\int_{\Omega} (1 - H(\phi(x, y))) dx dy} \quad (3.19)$$

The local minimisation of the Chan-Vese energy function is done by the gradient descent. It is assumed that the heaviside function is smoothed slightly to make it differentiable. Its derivative is the smoothed delta function: $\frac{dH(\phi)}{H(\phi)} = \delta(\phi)$. The gradient descent equation is calculated as:

$$\frac{\partial\phi}{\partial t} = \delta(\phi) \left(v \operatorname{div} \left(\frac{\nabla\phi}{|\nabla\phi|} \right) + (I(x, y) - h_2)^2 - (I(x, y) - h_1)^2 \right) \quad (3.20)$$

Similar to the Geodesic ACM, the Chan-Vese ACM minimises the energy function until the RoI object boundaries are reached. However, this is achieved independent of the gradient information but rather on the statistical information of the background and foreground (RoI) regions. This then allows Chan-Vese ACM to segment images even characterised by noise and smooth edges. Another significant advantage of the Chan-Vese ACM is that contours can be broken down into parts or joined together depending on the topology of the level set function. For these reasons, the Chan-Vese ACM is used in this work to segment and extract the weld joint as the RoI from the background of the thermite weld images.

3.3.2 Weld Joint Segmentation

The segmentation of the weld joint from the thermite weld images is based on the Chan-Vese ACM, which segments an image without relying on edge information. To segment the weld joint from the background image, Equation 3.17 is minimised with respect to the h_1 , h_2 and ϕ . For a fixed ϕ , the optimal values of h_1 and h_2 are calculated using Equations 3.18 and 3.19, respectively. For fixed values of h_1 and h_2 , the evolution of ϕ is calculated using Equation 3.20. Figure 3.3a and 3.3b shows the application of the Chan-Vese ACM and the segmented weld joint image, respectively.

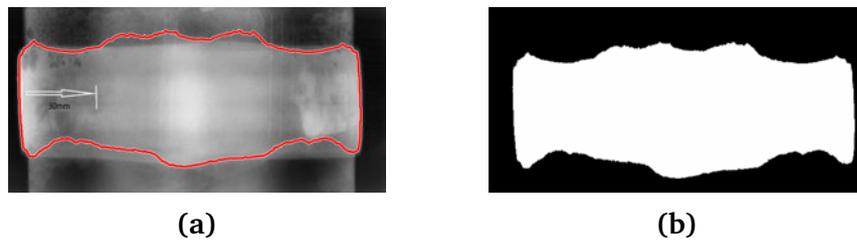


Figure 3.3: Image segmentation: (a) Application of Chan-Vese ACM and (b) segmented image.

3.3.3 Weld Joint Extraction

To extract the weld joint as the RoI, first, the coordinates and size of the foreground pixels from the segmented images must be obtained. Then, the obtained coordinates must be superimposed on the original image to allow for placement of the bounding box on the region of the weld joint (see Figure 3.4a). The area inside this bounding box is then cropped, and the final result is the image depicting only the weld joint without the background information (see Figure 3.4b).

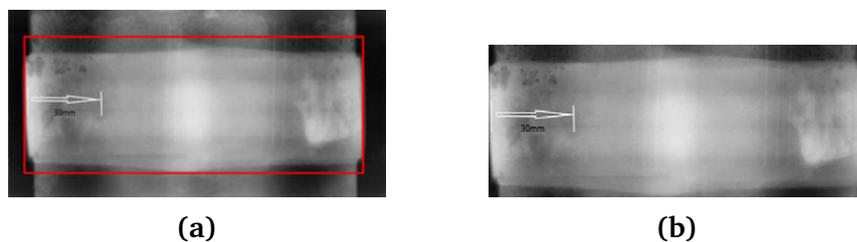


Figure 3.4: RoI extraction: (a) Image with bounding box and (b) Cropped image.

3.3.4 Post-Processing using Morphological Operations

Figure 3.5b shows that some images segmented by the proposed Chan-Vese ACM require to be post-processed in order to remove the defective regions (dark regions) inside the weld joint RoI. This will then allow the weld joint to be segmented accurately from the image background. In image processing, morphological operations are techniques that modify images based on shape. Some applications of morphology include texture analysis, noise elimination and boundary extraction [104]. When used in binary images, morphological operations aim to remove unwanted pixel information on the foreground and background regions of the image. This is achieved using a structuring element known as the kernel. The kernel is applied in all the possible areas of the segmented image to produce a post-processed image of a similar size. A pixel in the input binary image (1 or 0) is assigned a value of 1 only if all the pixels within the kernel have a value of 1; otherwise a pixel in the input image is assigned a value of 0. This means the dark pixels in the foreground region of the input image will become white in the post-processed image, and the white pixels in the background of the input image will become dark in the post-processed image.

Mathematical morphology operation used in this work is dilation. Dilation adds pixels on the boundaries of the image. In this work, dilation is used to add foreground pixels such that dark regions in the weld joint are eliminated (see Figure 3.5c). Dilation causes the white region (foreground pixels) to grow in size; thus, dark regions become smaller and smaller. Dilation takes two parameters as inputs; the first parameter is the input image to dilate, and the second parameter is the kernel. Given A as a set of input image coordinates, B as a set of kernel coordinates and B_x as a translation of B such that its origin is at x . Then, the Dilation of A by B is the set of all points in x such that the intersection of A with B_x is not empty. This is mathematically defined as [105]:

$$A \oplus B = \{x | (B)_x \cap A \neq \phi\} \quad (3.21)$$

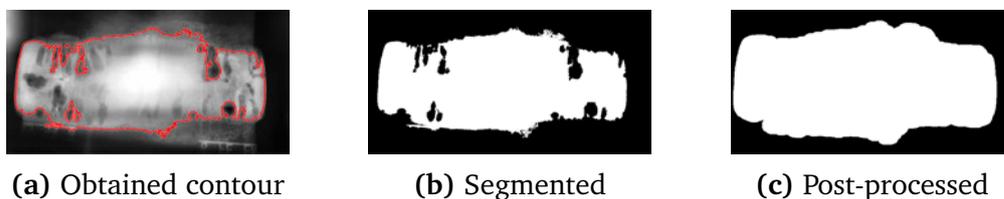


Figure 3.5: Post processing of the segmented image

3.3.5 Algorithm for Weld Joint Segmentation and ROI extraction

Algorithm 2 lists the steps used in this work for extracting the weld joint from the background of thermite weld images.

Algorithm 2 Image segmentation and weld joint (ROI) extraction

Require: Enhanced thermite weld images

Output: Weld joint images

- 1: **for** each image I in the dataset **do**
 - 2: Initialise ϕ .
 - 3: set n number of iterations.
 - 4: **for** $n=1$ to maximum n **do**
 - 5: **while** contour is not stationary **do**
 - 6: Calculate $h_1(\phi)$ and $h_2(\phi)$ using Equations 3.18 and 3.19 respectively.
 - 7: Evolve ϕ^{n+1} using Equation 3.20.
 - 8: **end while**
 - 9: **end for**
 - 10: Obtain the coordinates of the segmented foreground pixels.
 - 11: Impose to the original image.
 - 12: Place the bounding box across the coordinates.
 - 13: Crop the area inside the bounding box.
 - 14: save the cropped image.
 - 15: **end for**
-

3.4 Feature Extraction

The previous sections of the current chapter have provided the methods used in this work for image enhancement and weld joint (ROI) extraction. This section of the chapter introduces the feature extraction algorithms that aim to achieve automated weld defect classification while being robust to numerous image transformations. As discussed in Chapter 2, feature extraction techniques are divided into global and local feature extraction. Global features are computed by considering the entire image, and they represent the contents of the image by using a single vector; most shape and texture descriptors fall under this category.

Global features are attractive because they produce a compact representation of images where each image corresponds to a single point in a high dimensional feature space. Furthermore, global features are faster to compute and require less computational cost than local features. Some of the commonly used global feature extraction methods include the Histogram of Oriented Gradients (HoG) descriptor and the Grey

Level Co-occurrence Matrix (GLCM). However, features produced by the aforementioned techniques are not invariant to significant image transformations such as image rotation, scale and illumination changes and are sensitive to clutter and occlusion [106]. As a result, it is either assumed that an image contains only a single object or that good segmentation of the object from the image background is available.

The approach to overcoming these limitations, as stated by Lisin et al. [79] is to segment images into several regions, with each region representing a single object. However, image segmentation is by itself a challenging task that requires a high-level understanding of the image content. The limitations of global feature extractors are overcome by the methods based on local feature extraction. Local feature extractors find interesting characteristics of the image content despite significant changes in illumination, occlusion, viewpoint and clutter, and the image does not need to be segmented. A local feature is computed over a relatively small region of the image; therefore, it can be defined as a pattern in an image that differs from its immediate neighbourhood [107].

Two local feature extraction techniques are empirically compared for extracting defect features in the weld joint images; these are the Local Binary Patterns (LBP) descriptor and the Speeded-Up Robust Features (SURF) descriptor. In this work, the feature extractor is required to extract illumination and rotation invariant features, and both the LBP and SURF descriptors meet these requirements. It should be mentioned that the output from the SURF descriptor are keypoints that are highly discriminative, where many keypoint descriptor vectors represent each image; this then makes it challenging to train a classifier as the classification results will be impacted by outliers and the computational cost to train the classifier will be expensive. To address these challenges, the Bag of Visual Words (BoVW) approach is used in this work to cluster the keypoints into groups called visual words or the codewords; then, every weld joint image is represented by a global vector which is a count of the number of occurrence of each codeword on a given image.

This section is structured as follows. The mathematical approach, parameter selection and algorithm implementation of the LBP and SURF descriptors is presented in Subsections 3.4.1 and 3.4.2, respectively. Thereafter, the representation of SURF features using the BoVW approach is outlined in Subsection 3.4.3. Inputs are the extracted weld joint images, and the outputs are feature vectors extracted by the feature extraction techniques. The extracted features are stored for training and validating the classification algorithms.

3.4.1 Feature Extraction using Local Binary Patterns

The LBP descriptor is an effective descriptor for a simple way of extracting texture in an image. It operates by dividing an image into cells, then every pixel in a cell is thresholded with its neighbouring pixels, and the value is a binary number. The LBP value of pixel c surrounded by P neighbouring pixels placed on a circle of radius R from c is defined as [108]:

$$LBP_{(R,P)}(c) = \sum_{i=0}^{P-1} S(g_i - g_c) \times 2^i \quad (3.22)$$

Where g_c represents the grey intensity value of c and g_i represents the grey intensity value of circular symmetric neighbourhood pixels of c . The sign function $S()$ ensures that the LBP descriptor is invariant to illumination change; it is defined as:

$$S(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \quad (3.23)$$

The binary number generated is converted into a decimal number which forms the LBP code for the given centre pixel. Figure 3.6 illustrates the generation of the LBP code for the centre pixel highlighted by a red colour.

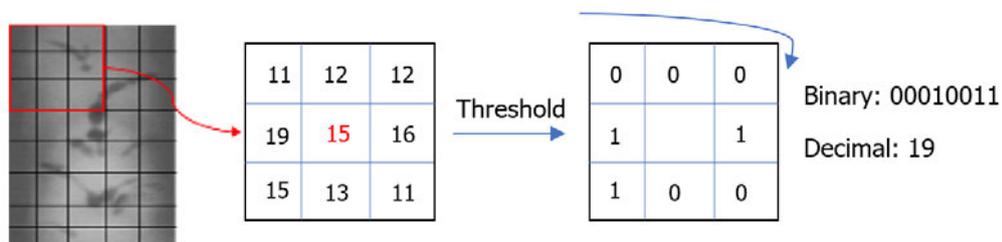


Figure 3.6: Computation of the LBP code

Assuming that the cell in Figure 3.6 has dimensions $N \times M$, the LBP code is computed for every pixel, and it is characterised by the distribution of codes representing the cell as the LBP histogram vector defined as:

$$z(k) = \sum_{i=1}^N \sum_{j=1}^M g(LBP_{(R,P)}(i, j) - k) \quad (3.24)$$

The original LBP descriptor of Equation 3.22 is not invariant to rotation, and it produces 2^P different binary codes produced from the neighbouring pixels. However, each neighbourhood pixel will move accordingly along the circle's parameter for a

rotated image, therefore yielding a different LBP value. A rotation invariant LBP descriptor is achieved by grouping together the LBP patterns that are the rotated versions of the same pattern. The rotation invariant LBP descriptor is formally defined as:

$$LBP_{(R,P)}^{ri} = \min\{ROR(LBP_{(R,P)}|i = 0, 1, \dots, P - 1)\} \quad (3.25)$$

Where the function $ROR(x, i)$ perform the circular i step bitwise right shift on the pattern binary string x for i number of times. The minimum between the resulting numbers is then selected. Keeping only the rotationally invariant patterns leads to a reduction in feature dimensionality, but the number of LBP codes increases drastically with an increase in P .

The extended LBP descriptor [109] uses uniform patterns to reduce the number of LBP codes. Uniform patterns have been experimentally shown to occur more frequently in texture images than non-uniform patterns. A pattern is said to be uniform if it contains at most two bitwise transitions from 0 to 1 or vice versa when the binary string is considered circular. For example, 0001000 is a uniform pattern because it has two transitions, while 0101010 is not a uniform pattern because it has 6 transitions. To distinguish between the uniform and non-uniform patterns, the uniformity measure U is introduced, and it counts the number of spatial transitions (1's and 0's) between successive bits in the circular representation of the pattern binary code. U is defined as:

$$U(LBP_{(R,P)}) = |S(g_{i-1} - g_c) - S(g_0 - g_c)| + \sum_{i=0}^{P-1} |S(g_{i-1} - g_i) - S(g_i - g_{i+1})| \quad (3.26)$$

All patterns with $U > 2$ (more than two spatial transitions) are called non-uniform otherwise patterns are called uniform. The modified rotation invariant uniform LBP descriptor is then defined as:

$$LBP_{(R,P)}^{riu} = \begin{cases} \sum_{i=0}^{P-1} S(g_i - g_c) \times 2^i, & \text{if } U(LBP_{(R,P)}) \geq 2 \\ P+1, & \text{Otherwise} \end{cases} \quad (3.27)$$

To construct a histogram feature vector in a cell of a given image using uniform patterns, the LBP descriptor dedicates one bin (bin 0) of the histogram for the accumulation of non-uniform patterns while the remaining bins are for uniform patterns. For example, using (8,1) neighbourhood gives a total combination of 256 possible patterns, of which 58 are uniform. This then gives a total of 59 patterns which is a significant reduction in the feature vector length compared to the LBP descriptor without uniform patterns. The histogram of all cells in an image are normalised and concatenated to form a final feature vector, as shown in Figure 3.7.

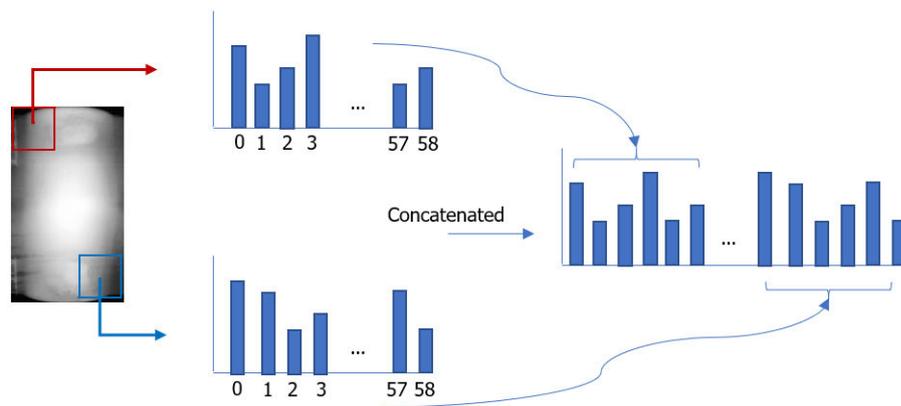


Figure 3.7: Computation of a histogram vector using uniform LBP

Parameter selection

The LBP descriptor has several parameters, some of which requires fine-tuning to achieve the best feature classification results. Parameters such as the number of neighbours, the radius and the cell size are usually the main parameters which are optimised for best results depending on the type of LBP descriptor used [110, 111, 112]. In this work, the rotation invariant uniform LBP descriptor is used, thus the number of neighbours and the radius are kept at (8,1) since the uniform patterns are found to occur more frequently on this combination [109]. Keeping the number of neighbours at 8 is also a great choice for avoiding a long feature vector. The LBP parameter which is optimised here is the cell size. Table 3.1 lists the LBP parameters used in this work.

Table 3.1: LBP Parameters

| Parameter | Value |
|---------------------|-----------------------------------|
| Number of neighbors | 8 |
| Radius | 1 |
| Cell size | Optimal cell size to be evaluated |
| Number of bins | 59 |
| Normalisation | L2-norm |

Algorithm for Extracting Features using LBP

Algorithm 3 gives the step involved for extracting features in weld joint images using the LBP descriptor.

Algorithm 3 Feature extraction using LBP

Require: Weld joint images from training and validation dataset

output: Concatenated feature vector z per image

- 1: **for** each image I in the dataset **do**
 - 2: Divide into cells.
 - 3: **for** each cell **do**
 - 4: Compare every pixel's value to its P neighbours.
 - 5: Threshold every pixel and obtain a binary number using Equation 3.22.
 - 6: Obtain rotation-invariant uniform LBP using Equation 3.27.
 - 7: Compute and normalise the histogram.
 - 8: **end for**
 - 9: Form a feature vector z .
 - 10: **end for**
-

3.4.2 Feature Extraction using Speeded Up Robust Features

Keypoint extraction using the SURF descriptor is broken down into four steps; keypoint detection, keypoint localisation, orientation assignment and keypoint description. This subsection presents the mathematical approach of each step and the algorithm used in this work to extract keypoints using the SURF descriptor.

Keypoint Detection

The SURF descriptor uses the Hessian matrix to determine the location and scale of the potential keypoints. For a given pixel (x, y) in image I , the Hessian matrix is calculated as:

$$H(I(x, y)) = \begin{bmatrix} \frac{\partial^2 I}{\partial x^2} & \frac{\partial^2 I}{\partial x \partial y} \\ \frac{\partial^2 I}{\partial x \partial y} & \frac{\partial^2 I}{\partial y^2} \end{bmatrix} \quad (3.28)$$

To achieve scale-invariant keypoints, SURF searches for potential keypoints in an image at various Gaussian scales and octaves. Thus, for a given point $c = (x, y)$ at Gaussian scale σ in an octave, the Hessian matrix $H(c, \sigma)$ is defined as:

$$H(c, \sigma) = \begin{bmatrix} L_{xx}(c, \sigma) & L_{xy}(c, \sigma) \\ L_{xy}(c, \sigma) & L_{yy}(c, \sigma) \end{bmatrix} \quad (3.29)$$

Where $L_{xx}(c, \sigma)$, $L_{yy}(c, \sigma)$ and $L_{xy}(c, \sigma)$ are the convolutions of Gaussian second-order derivatives with image I at point c in x , y and xy directions, respectively. The determinant of Hessian matrix at this location is then defined as [113].

$$\det(H_{app}) = D_{xx}D_{yy} - (0.9D_{xy})^2 \quad (3.30)$$

Where D_{xx} , D_{yy} and D_{xy} are the approximations of the Gaussian second-order derivatives in x , y and xy directions respectively. The SURF algorithm uses responses of box filters to approximate these three derivatives in respective directions. Three such box filters are depicted in Figure 3.8.

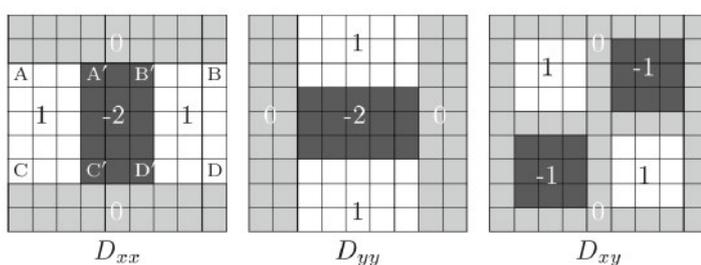


Figure 3.8: Three box filter approximation of the second-order derivatives of Gaussian filters [3]

The box filter responses are computed using the integral images. In an integral image, the value of any pixel (x, y) is the sum of all the pixel values above and to the left of the same pixel location in the original image. The integral image I_{int} computed from image I can be calculated as [114]:

$$\sum I_{int} = \sum_{x' \leq x, y' \leq y} I(x', y') \quad (3.31)$$

The concept of integral images allows for quick and efficient computation of box filters. For example, the approximation of D_{xx} filter in Figure 3.8 is calculated by first calculating the area enclosed by region $A'B'C'D'$ and then subtracting the area enclosed region by $ABCD$. These area calculations can be carried out efficiently using integral images.

Scale Space Representation

To be invariant to scale changes, scale-space representation in SURF is achieved by upscaling the filter sizes rather than iteratively reducing the image size [115]. The filter sizes for various octaves and layers is depicted in Figure 3.9. The 9×9 box filter is the initial scale layer, and it corresponds to the approximation of Gaussian second-order derivative at $\sigma = 1.2$.

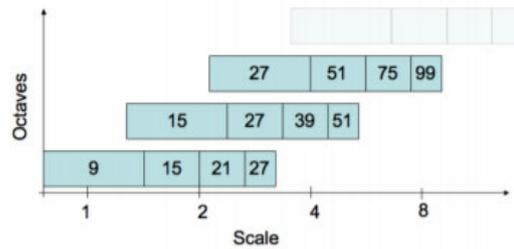


Figure 3.9: The scale, the filter sizes and octaves in SURF [4]

Keypoint Localisation

The determinant of the Hessian matrix determines the potential keypoint, but some are weak and need to be eliminated; this is done in the keypoint localisation step. Keypoint localisation is achieved in three respective stages; in the first stage, all keypoints within an octave are tested against a fixed threshold value. Keypoints above the threshold value are accepted and passed on to the second stage. Keypoints below the threshold are discarded. The second stage is non-maximum suppression in a $3 \times 3 \times 3$ neighbourhood. In this stage, every keypoint is compared to its 26 neighbouring pixels, 9 in the scale below and above it and 8 in the current scale. A keypoint is considered a strong keypoint if its value is higher or lower than all its neighbours (see Figure 3.10).

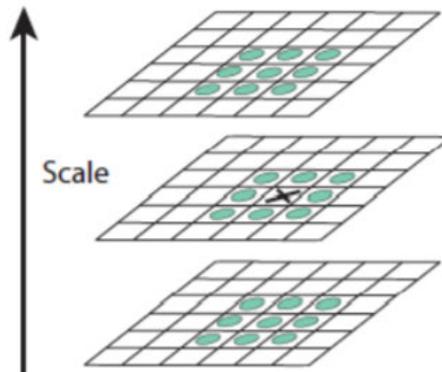


Figure 3.10: Non maximum suppression [5]

The last step of keypoint localisation is to interpolate the nearby data to determine the position and scale of keypoints to sub-pixel accuracy. This is achieved by fitting a 3D quadratic function around the neighbourhood of each local extrema, and its peak value is selected as a sub-pixel and sub-scale location. The function is approximated by Taylor expansion of the scale-space function $D(x, y, \sigma)$ with the keypoint (from

stage 2) as the origin. The Taylor expansion is defined as:

$$D(z) = D + \frac{\partial D^T}{\partial z} z + \frac{1}{2} z^T \frac{\partial^2 D}{\partial z^2} z \quad (3.32)$$

Where D and its derivatives are evaluated at the keypoint candidate $z_0 = [x_0, y_0, \sigma_0]^T$ and offset point $z = [\delta x, \delta y, \delta \sigma]^T$. The location of the extrema is then evaluated by setting the derivative of Equation 3.32 to 0, yielding:

$$\hat{z} = \frac{\partial^2 D^{-1}}{\partial z^2} \frac{\partial D}{\partial z} \quad (3.33)$$

Orientation Assignment

The output from the previous step is the scale-invariant keypoints which are localised to a sub-pixel accuracy in terms of (x, y, σ) . The orientation assignment step aims to achieve rotation invariant keypoints by assigning to each a reproducible orientation information. This is done in two steps; first, a circular region of radius 6σ is taken around every keypoint, and within this region, Haar wavelets responses of size 4σ in x and y directions are calculated. The obtained responses are then weighted using a Gaussian kernel centred around every keypoint and plotted as vector points in x and y coordinates. In the second step, a window of size $\pi/3$ is rotated around a keypoint. Points inside this window are then summed up, and the most dominant results are assigned as the orientation of the keypoint. The orientation assignment step can be understood by referring to Figure 3.11.

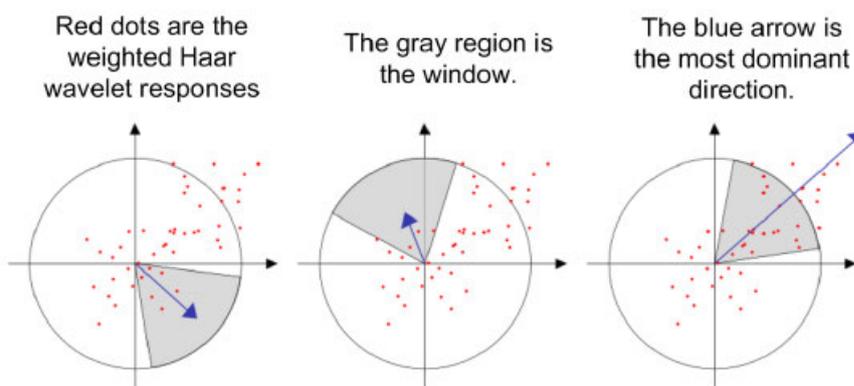


Figure 3.11: Keypoint orientation assignment [6]

Keypoint Description

This step constructs a square region centred around the keypoint and oriented along the dominant orientation. This region is divided into sub-regions of size 4×4 , and

for each sub-region, Haar wavelets responses are calculated at 5×5 regular spaced sample points. The x and y wavelet responses denoted by d_x , and d_y , respectively are calculated and summed up to give a first entry to the feature vector. Absolute values of the responses $|d_x|$ and $|d_y|$ are also added to the feature vector to obtain the information on the polarity of the intensity changes. Thus for each sub-region, a vector is four-dimensional given as:

$$v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|) \quad (3.34)$$

Since there is a total of 16 sub-regions within the square region, the SURF descriptor across every keypoint is, therefore, a 64-dimensional feature vector.

Algorithm for Extracting Features using SURF

Algorithm 4 presents the steps used to extract features using the SURF descriptor.

Algorithm 4 Keypoint detection and description using SURF

Require: Weld joint images from training and validation dataset

Output: Keypoint descriptor vectors per image

- 1: **for** each image I in the dataset **do**
 - 2: Calculate integral image I_{int} using Equation 3.31.
 - 3: Construct scale space.
 - 4: **for** each pixel in I_{int} **do**
 - 5: Calculate D_{xx} , D_{yy} , D_{xy} using box filters.
 - 6: Calculate the Hessian determinant $H_{(c,\sigma)}$ using Equation 3.30.
 - 7: **if** $H_{(c,\sigma)} \geq \text{minHessian}$ **then**
 - 8: Store as a potential keypoint
 - 9: **end if**
 - 10: **end for**
 - 11: **for** each potential keypoint **do**
 - 12: Perform non-maxima suppression in a $3 \times 3 \times 3$ neighbourhood.
 - 13: Find sub-pixel and scale location using Equation 3.32.
 - 14: Assign orientation.
 - 15: Construct a square region centred around a keypoint.
 - 16: Divide into 4×4 sub-regions.
 - 17: **for** each sub region **do**
 - 18: Calculate a keypoint descriptor vector using Equation 3.34.
 - 19: **end for**
 - 20: Concatenate and store a keypoint vector.
 - 21: **end for**
 - 22: **end for**
-

3.4.3 Image Representation using Bag of Visual Words

The SURF descriptor produces many descriptor vectors for each image. This is not ideal for training a classifier because the results will be affected significantly by the outliers, and the computation cost will be high. Thus, this work proposes the BoVW approach to represent each image using only a single feature vector. BoVW is achieved in three steps. Codebook construction, coding and pooling. As shown in Figure 3.12, the codebook is constructed by grouping together the SURF keypoint descriptors from the training dataset that are similar, and each group represent a visual word or a codeword. Coding and pooling represent every image in the dataset as a global feature vector which is a count of the number of times each codeword appears on an image.

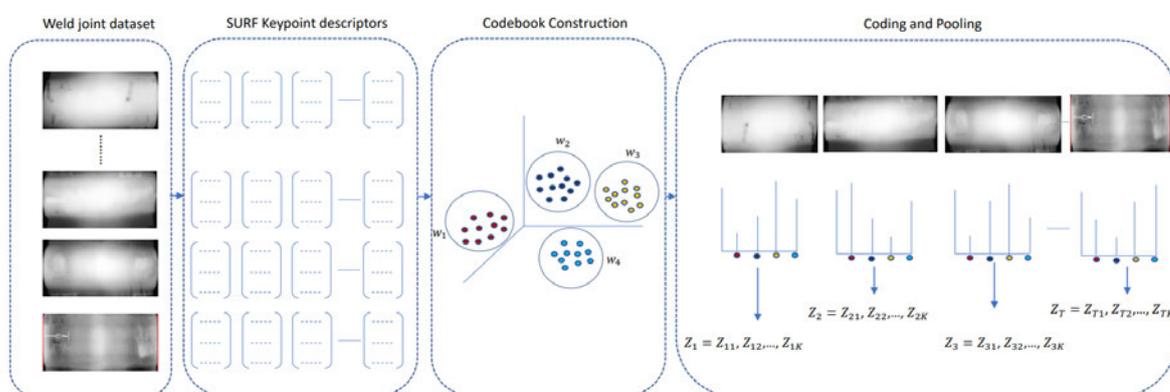


Figure 3.12: Image representation using the bag of visual words

Codebook Construction

In the codebook construction step, all the keypoint descriptor vectors from the training dataset are clustered together, and each cluster represents a codeword. Let $V = \{v_j \mid j = 1, 2, \dots, N\}$ be a set of unordered SURF keypoint descriptors extracted from the training dataset where $v_j \in \mathbb{R}^D$ is a keypoint descriptor vector and N is the total number of keypoint descriptors. In this work, the K means clustering algorithm is used to construct the codebook. This is done by clustering the N keypoint descriptor vectors into K clusters. The output from K means clustering is then a codebook defined as $C = \{c_k \mid k = 1, 2, \dots, K\}$ where $c_k \in \mathbb{R}^D$ is the mean vector of the k^{th} cluster.

Coding

The coding step aims to represent every image in the dataset in terms of the codebook elements (codewords). The coding step can be modelled using the function f defined

as:

$$\begin{aligned} f : \mathbb{R}^D &\longrightarrow \mathbb{R}^K \\ v_j &\longrightarrow \beta_j \end{aligned} \quad (3.35)$$

where $\beta_j = \{(\beta_{k,j}) | k = 1, \dots, K\}$ maps a descriptor vector v_j into the closest codeword c_k in the codebook according to the following hard coding equation

$$\beta_{k,j} = \begin{cases} 1, & \text{if } k = \arg \min_{k \in \{1, \dots, K\}} \|v_j - c_k\|_2^2 \\ 0, & \text{otherwise} \end{cases} \quad (3.36)$$

Where $\beta_{k,j}$ is the k^{th} component of the encoded vector β_j .

Pooling

The final step in the BoVW approach is to construct a vector z that provides a global description of an image. This vector is a count of how many times each codeword appears on a given image. The idea of the pooling step is to concatenate and add all the elements of the encoded descriptor vector for every keypoint on an image. Thus, given an image with the total number of n descriptors, the k^{th} component of vector z is calculated as:

$$z_k = \sum_{j=1}^n \beta_{k,j} \quad (3.37)$$

Parameter Selection

Two important parameters of the BoVW approach with SURF descriptor (hereafter referred to as "BoSURF approach"), which requires fine-tuning for optimal classification results, are found in the keypoint detection, description and the codebook construction steps, respectively. In the keypoint detection and description step, the SURF descriptor computes the keypoints using the determinant of the Hessian matrix. Then it removes some by thresholding against a fixed Hessian determinant value minHessian . Though minHessian is based on heuristics, the optimal value has been found to be between 400 and 800 in several research studies [116, 117]. Therefore, minHessian of 500 is used in this work. The other parameter of interest is in the codebook construction step where keypoints are clustered into K number of clusters, and each cluster represents a codeword, again the question of which K to use is also based on heuristics; therefore, different values of K ranging from 200 to 2000 are experimented in this work for optimal results.

Algorithm for representing SURF Features using BoVW approach

The steps for representing the weld joint images using the BoSURF approach are illustrated by Algorithm 5.

Algorithm 5 Image representation using BoSURF approach

Require: Unlabeled keypoint descriptor vectors

Output: Global feature vector z per each weld joint image

- 1: Store the training descriptor vectors into a dictionary V .
 - 2: Randomly choose K number of descriptor vectors from V to form codeword centres.
 - 3: **while** codeword centres are unchanged **do**
 - 4: Allocate each descriptor vector to the nearest codeword centre.
 - 5: Replace codeword centres with the mean of the descriptor vectors in their codewords.
 - 6: **end while**
 - 7: Form a codebook C with K number of codewords.
 - 8: **for** every image I in the dataset **do**
 - 9: **for** every descriptor vector in I **do**
 - 10: Assign to the nearest codeword in C using Equation 3.35.
 - 11: **end for**
 - 12: Form a histogram feature vector z using Equation 3.37.
 - 13: **end for**
-

3.5 Feature Classification

The feature vectors extracted using the techniques described in the preceding section of the chapter are independently used as inputs to train and validate each classification algorithm. The objective of the classifier is to learn the decision boundary that can separate new or unseen weld joint images into one of four weld defects types considered in this work. As discussed in Chapter 2, several classifiers have been used for multiclass defect classification in radiography images. It should be noted that the dataset used to conduct experiments in this work is very limited. Training a classifier based on the limited dataset results in a classifier overfitting as it has more degrees of freedom to construct a decision boundary. Another challenge arises when the dataset is characterised by outliers. Outliers have a significant impact on the small dataset as they can significantly skew the decision boundary.

Three classification algorithms deemed effective for modelling small dataset are used in this work to address the aforementioned challenges [118, 119, 120]. These are

the Support Vector Machines (SVM), the K-Nearest Neighbours (K-NN) and the Naive Bayes classifiers. Therefore, this section of the chapter provides a detailed explanation of the mathematical approach and algorithm implementation of the considered classification algorithms for classifying thermite weld defects.

3.5.1 Feature Classification using Support Vector Machines

SVM are one of the widely used classification algorithms due to many promising characteristics in terms of performance. SVM were originally formulated for a binary classification task, but their use soon extended for multiclass problems through one vs one and one vs many SVM. This subsection provides the mathematical theory of SVM and how they are used in this work; at the end of the subsection, an algorithm detailing the classification of weld joints images using SVM is presented.

Mathematical Approach

For a binary classification task, let $((v_1, y_1), \dots, (v_n, y_n))$ be the training dataset where v_i are the feature vectors representing the samples and $y \in (-1, +1)$ are the corresponding class labels for the samples. With reference to Figure 3.13, SVM is a learning algorithm that attempts to find the hyperplane that separates the positive samples (+1 labelled) from the negative samples (-1 labelled) with the largest margin, where w is the vector constrained to be perpendicular to the hyperplane, b is the bias term, and $b/\|w\|$ is the perpendicular distance from the origin to the hyperplane. The margin of the hyperplane is defined as the shortest distance between the positive and negative samples, which are known as support vectors. For all the samples in the training dataset, the following constraints must be satisfied.

$$w.v_i + b \geq +1 \quad \text{for } y_i = +1 \quad (3.38)$$

$$w.v_i + b \leq -1 \quad \text{for } y_i = -1 \quad (3.39)$$

Samples for which Equations 3.38 and 3.39 hold lie on the hyperplanes H_1 and H_2 of Figure 3.13. It turns out the two constraints can be combined together and be represented as:

$$y_i(w.v_i + b) \geq +1 \quad \forall i \quad (3.40)$$

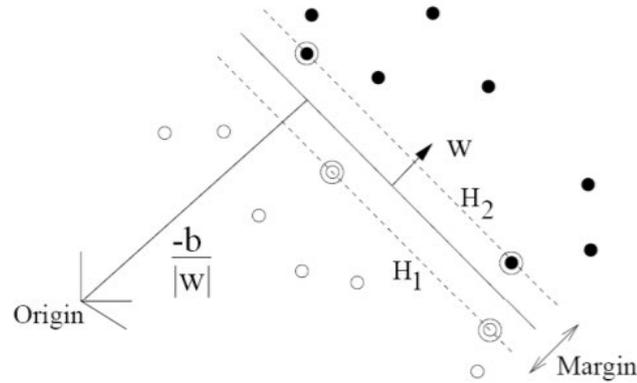


Figure 3.13: A hyperplane that separates the negative and positive samples [7]

By referring to Figure 3.13, the margin which must then be maximised can be computed as the distance between H_1 and H_2 planes defined as:

$$d = \frac{|1 - b|}{\|w\|} - \frac{|-1 - b|}{\|w\|} = \frac{2}{\|w\|} \quad (3.41)$$

Thus the margin which must be maximised for an optimal separating hyperplane is equivalent to solving a primal optimisation problem defined as:

$$\min \frac{1}{2} \|w\|^2 \quad \text{subject to} \quad y_i(w \cdot v_i + b) \geq +1 \quad \forall i \quad (3.42)$$

To find the maxima or minima of any function without having to worry about the constraints, Lagrangian formulation is used. It introduces new Lagrangian multiplier α_i for each constraint and the minimisation problem of Equation 3.42 becomes:

$$L = \frac{1}{2} \|w\|^2 - \sum_{i=1}^l \alpha_i y_i (x_i w_i + b) + \sum_{i=1}^l \alpha_i \quad (3.43)$$

Taking the partial of derivative of Equation 3.43 with respect to the vector w and the bias b yields:

$$w = \sum_{i=1}^l \alpha_i y_i v_i \quad (3.44)$$

$$\sum_{i=1}^l \alpha_i y_i = 0 \quad (3.45)$$

The expression of Equation 3.44 defines the vector w as the linear sum of some of the samples in the dataset. Substituting Equation 3.44 and Equation 3.45 into Equation 3.43 gives the formulation of the dual SVM defined as:

$$L = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (v_i \cdot v_j) \quad \text{subject to} \quad \sum_{i=1} \alpha_i y_i = 0 \quad \text{and} \quad \alpha_i \geq 0 \quad (3.46)$$

By solving the dual optimisation problem, the coefficients α_i are found. The samples with $\alpha_i > 0$ are called the support vectors, and they lie on H_1 and H_2 hyperplanes. Only the support vectors affect the solution of the SVM problem; hence only the support vectors are needed to express the solution of the vector w . The decision rule for classification of the new, unseen sample represented by the vector z is therefore defined as:

$$f(z) = w^T z_i + b \quad (3.47)$$

Which is equivalent to:

$$f(z) = \sum_i^M y_i \alpha_i (v_i^T \cdot z) + b \quad (3.48)$$

The predicted class label of vector z is then determined by the sign of the decision function stated above. The formulation of the SVM classifier discussed till now assumes the training samples are linearly separable. In a real-life classification task, however, the data is characterised by the presence of noise and outliers; thus, data samples cannot be separated linearly. Soft margin SVM tackle this problem by introducing a *slack variable* ξ_i which allows some samples to lie amongst samples of the opposite class. The primal optimisation problem of Equation 3.42 taking into account ξ_i is then defined as:

$$\min \frac{1}{2} \|w^2\| + C \sum_i \xi_i \quad \text{subject to} \quad y_i (w \cdot x_i + b) \geq +1 - \xi_i \quad \forall \xi_i \quad (3.49)$$

Where C is a parameter that controls the misclassification error, applying the Lagrangian formulation of Equation 3.49 and then taking the partial derivative with respect to w and b yields the dual formulation problem defined as:

$$L = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (v_i \cdot v_j) \quad \text{subject to} \quad \sum_{i=1} \alpha_i y_i = 0 \quad \text{and} \quad C \leq \alpha_i \leq 0 \quad (3.50)$$

The formulation of non linear SVM is possible for cases where the samples are non-linearly separable. The main idea is to transform samples to a high dimensional feature space χ where they can easily be separated. The transformation then requires the dot product between any pairs of samples to be computed in χ (i.e $\phi(v_i) \cdot \phi(v_j)$). This transformation is computationally expensive thus the kernel functions are used.

A kernel function K that corresponds to the dot product in χ is defined as: $K(v_i, v_j) = \phi(x_i) \cdot \phi(v_j)$, thus, only K is needed for computing the dot product without mapping into a high dimensional feature space. The dual problem of Equation 3.51 is then defined as:

$$L = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (K(v_i \cdot v_j)) \quad \text{subject to} \quad \sum_{i=1} \alpha_i y_i = 0 \quad \text{and} \quad C \leq \alpha_i \leq 0 \quad (3.51)$$

Some of the commonly used kernel functions are the linear, polynomial and Radial Basis Function (RBF). In many classification tasks, the linear and polynomial kernels have been found to require less computational cost, but they usually achieve low classification accuracy compared to the RBF kernel [121, 122, 123]. The RBF is used as the kernel function in this work, and it is defined as:

$$K(v_i, v_j) = \exp\left(-\frac{\|v_i - v_j\|^2}{2\sigma^2}\right) \quad (3.52)$$

To achieve a multi-classification of thermite weld defects, the one vs one SVM classifier is used, where two pairs of classes are trained at a time. Thus, a total of $D(D-1)/2$ classifiers are obtained, where D is the total number of classes. An unknown feature vector is assigned to a class label based on the majority vote.

Algorithm for Feature Classification using SVM

Algorithm 6 gives the steps involved in the classification of weld joint images using SVM.

Algorithm 6 Feature classification using SVM

Require: Training and validation feature vectors

Output: Class label for each feature vector in the validation dataset

- 1: Train a one vs one SVM model.
 - 2: **for** for any pair of classes y_i and y_j **do**
 - 3: Map the training feature vectors into higher space using a kernel function.
 - 4: Obtain a separating hyperplane by minimising Equation 3.51.
 - 5: **end for**
 - 6: **for** each feature vector z in the validation dataset **do**
 - 7: Assign to class label with the majority vote.
 - 8: **end for**
-

3.5.2 Feature Classification using the K-Nearest Neighbors

Feature Classification using the K-NN classifier is conceptually simple. Learning using K-NN consists of storing the training samples, and when a new query sample is made, a set of similar related samples is retrieved from the memory and used to assign the class label to the new query sample. This Subsection presents the mathematical approach and the algorithm for the classifying weld joint images using the K-NN classifier.

Mathematical Approach

To explain the workings of K-NN, let $((v_1, y_1), \dots, (v_n, y_n))$ be the weld joint training dataset where v_i are feature vectors representing the training samples in a high dimensional feature space \mathbb{R}^m , y_i are the class labels of the samples. The training phase of the K-NN stores the training samples and when the query sample represented by vector z from the validation data is made, the distance between the z and every other training sample is calculated. The distance measure used in this work is the Euclidean distance, for any two vectors v_i and v_j it can be defined as:

$$d(v_i, v_j) = \sqrt{\sum_{r=1}^m (a_r(v_i) - a_r(v_j))^2} \quad (3.53)$$

Then, k number of samples (v_1, v_2, \dots, v_k) which are nearest to z are used to assign the class label of z according to the equation defined as:

$$y(z) \leftarrow \arg \max_{c \in C} \sum_{i=1}^k \delta(c, y(v_i)) \quad (3.54)$$

Where $y(z)$ is the class of sample z , $c \in C$ is the class label and $\delta(c, y(z))$ is equal to 1 if c is equal to $c(v_i)$, otherwise $\delta(c, y(z))$ is equal to 0. One obvious disadvantage of assigning the class label based on the majority vote is that nearest k samples may vary widely in their distance, and the closest neighbours more reliably indicate the class label for the query sample. For these reasons, weighted K-NN is used in this work. In weighted K-NN, the contribution of each of the k nearest samples is weighted according to their distance from the query sample z , thus giving greater weight to the closest neighbours. By weighting the vote of each nearest sample, then Equation 3.54 becomes.

$$y(z) \leftarrow \arg \max_{c \in C} \sum_{i=1}^k w_i \delta(c, y(z)) \quad (3.55)$$

Where w_i is the weighting function. In this work, samples are weighted according to their inverse squared distance from z defined as:

$$w_i = \frac{1}{d(z, v_i)^2} \quad (3.56)$$

Algorithm for Feature Classification using K-NN

Algorithm 7 gives the steps involved in the classification of weld joint images using the KNN classifier.

Algorithm 7 Feature classification using KNN

Require: Training and validation feature vectors

Output: Class label for each feature vector in the validation dataset

- 1: Training phase:
 - 2: **for** each feature vector v in the training dataset **do**
 - 3: Save to a dictionary V
 - 4: **end for**
 - 5: Classification phase:
 - 6: **for** each feature vector z in validation dataset **do**
 - 7: Compute the distance between z and vectors in V using Equation 3.53.
 - 8: Assign the class label to z using Equation 3.55.
 - 9: **end for**
-

3.5.3 Feature Classification using Naive Bayes

Naive Bayes classifier belongs to a family of Bayesian networks, where the class assignment of an unknown sample is based on the class conditional probabilities, with each representing the probability that the unknown sample belongs to the respective class. The following subsections detail the mathematical approach of the Naive Bayes classifier. The algorithm of how Naive Bayes is used in this work to classify weld joint images is also presented.

Mathematical Approach

Given M classes y_1, y_2, \dots, y_M and an unknown sample represented by a feature vector z in a high dimensional feature space \mathbb{R}^m . The probability that z belongs to class y_i is defined according to Bayes rule as [124]:

$$P(y_i|z) = \frac{P(z|y_i)P(y_i)}{P(z)} \quad (3.57)$$

Where $P(y_i|z)$ is the probability that sample z belongs to class y_i , $P(z|y_i)$ is the probability of generating sample z given class y_i , $P(y_i)$ is the prior probability of class y_i and $P(z)$ is the probability of sample z occurring. Modelling $P(z|y_i)$ is impractical given z is a vector in a high dimensional feature space. Thus in Naive Bayes, it is assumed that individual $z_{i's}$ is conditionally independent given y . The numerator of Equation 3.57 then becomes:

$$P(z|y_i)P(y_i) = P(z_1|y_i) \cdot P(z_2|y_i) \dots P(z_m|y_i) \cdot P(y_i) = \prod_{k=1}^m P(z_k|y_i)P(y_i) \quad (3.58)$$

$P(z)$ is the same for all the classes, and it does not affect the decision. Thus, equation 3.58 simplifies to:

$$P(z|y_i) = \prod_{k=1}^m P(z_k|y_i)P(y_i) \quad (3.59)$$

$P(y_i)$ is the class prior probability. Given N number of feature vectors from the training dataset and N_i number of feature vectors which belongs to class y_i , then the prior probability is calculated as:

$$P(y_i) = \frac{N_i}{N} \quad (3.60)$$

To assign the class label to an unknown sample, the value of Equation 3.59 is computed for each class and the class where this value is maximal is selected. This is computed as y for sample z :

$$y \leftarrow \arg \max_{y_i} \prod_{k=1}^m P(z_k|y_i)P(y_i) \quad (3.61)$$

Algorithm for Feature Classification with Naive Bayes

Algorithm 8 presents the steps involved for the classification of weld joint images using the Naive Bayes classifier.

Algorithm 8 Feature classification using Naive Bayes

Require: Training and validation feature vectors

Output: Class label for each feature vector from the test dataset

- 1: Training phase:
 - 2: **for** each class y_i **do**
 - 3: Calculate prior probability using Equation 3.60.
 - 4: **end for**
 - 5: Classification phase:
 - 6: **for** each feature vector z in the validation dataset **do**
 - 7: Assign the class label using Equation 3.61.
 - 8: **end for**
-

3.6 Evaluation Methods

The preceding sections of the chapter have provided the mathematical approach and algorithms used in this work for feature extraction and classification. To measure how each algorithm perform when compared to others, evaluation matrices must be defined. The algorithm's performance is simply knowing how well the algorithm can correctly assign the validation feature vectors to the correct class labels. This section of the chapter provides such evaluation matrices used in this work to evaluate the developed algorithms.

3.6.1 Confusion Matrix

In this work, the performance of a classification algorithm is evaluated by computing four attributes; the number of correctly recognised class examples (true positives), the number of correctly recognised examples that do not belong to the class (true negative) and examples that either were not recognised as class examples (false negatives) or that were incorrectly assigned to the class (false positives). The four attributes yield a confusion matrix which is depicted in Table 3.2 for a binary classification task.

Table 3.2: Confusion matrix table

| | Predicted positive | Predicted negative |
|-----------------|-------------------------|-------------------------|
| Actual positive | True positives(tp) | False negatives(fn) |
| Actual negative | False positives(fp) | True negatives (tn) |

Three performance measures can be computed from the confusion matrix, namely the average accuracy, precision, and recall. The average accuracy is calculated as the

total number of correctly recognized validation examples divided by the total number of examples in the validation dataset. Precision is calculated as the total number of correctly recognised positive examples divide by the number of examples labelled by the classifier as positive. The recall is calculated as the total number of correctly recognised positive examples divide by the total number of positive examples in the validation dataset. Equations 3.62 - 3.65 defines the performance measures for a multi-class classification task. For a single class C_i the performance measure is defined by tp_i , fn_i , fp_i and tn_i . Where l is the total number of classes in the validation dataset.

$$\text{Average accuracy} = \frac{\sum_{i=1}^l \frac{tp_i + tn_i}{tp_i + fn_i + fp_i + tn_i}}{l} \quad (3.62)$$

$$\text{Error rate} = 1 - \frac{\sum_{i=1}^l \frac{tp_i + tn_i}{tp_i + fn_i + fp_i + tn_i}}{l} \quad (3.63)$$

$$\text{Precision} = \frac{\sum_{i=1}^l tp_i}{\sum_{i=1}^l tp_i + fp_i} \quad (3.64)$$

$$\text{Recall} = \frac{\sum_{i=1}^l tp_i}{\sum_{i=1}^l tp_i + fn_i} \quad (3.65)$$

3.6.2 K Fold Cross-Validation

Cross-validation is a method used to evaluate the performance of the classification algorithm by measuring how well the algorithm generalises to new data. Before training, some of the samples in the dataset are removed, and once training has completed, the removed samples are used to evaluate the performance of the trained model. Many techniques can be used to perform cross-validation. The simplest technique is called the holdout method. In this technique, the data is separated into two datasets, namely the training dataset and validation dataset. The training dataset is used to train the model, and the test set is used to evaluate the model. However, there is a possibility of high biasness with the holdout method for limited data. K fold cross-validation is a technique to improve the holdout method. It ensures that every observation from the original dataset has a chance to appear on the training and validation datasets, thus resulting in a less biased model. The original dataset is randomly divided into K number of folds. The first fold is removed for evaluating the model, and the model is trained on the remaining $K - 1$ folds. Then the second fold is removed from the dataset, and the model is trained using the first and the last $K - 2$ folds. The process is repeated K number of times, and each time the results are reported. This thesis work uses the 5 fold cross-validation technique to evaluate

the performance of the thermite weld defect classification models.

3.7 Conclusion

This chapter has introduced the mathematical approaches and algorithms used in this work to classify defects in thermite weld images. The methods were presented in terms of image enhancement, RoI extraction, feature extraction and feature classification. Thermite weld image enhancement was carried out based on the CLAHE technique, and the Weld joint RoI extraction was achieved using the Chan-Vese ACM. For feature extraction, two techniques were proposed for comparison: the LBP descriptor and the BoVW approach with SURF descriptor (BoSURF). It was further outlined that specific parameters require fine-tuning for optimal results. In this work, the LBP cell size parameter on the LBP descriptor and the codebook size on the BoSURF approach are fine-tuned. Subsequently, the performance of feature extraction techniques was evaluated using the SVM, K-NN and Naive Bayes classifiers. The next chapter presents the experimental results and discussion.

Chapter 4

Experimental Results and Discussion

4.1 Introduction

This chapter aims to provide a detailed presentation of the results obtained from conducting the experiments using methods presented in Chapter 3. It is structured as follows: Dataset for the experiments is described in Section 4.2, followed by the presentation of the results obtained after image enhancement and weld joint ROI extraction algorithms in Sections 4.3 and 4.4, respectively. Then, the classification results obtained from the Local Binary Patterns (LBP) descriptor are presented in Section 4.5. Subsequently, the classification results obtained from the Bag of Speeded Up Robust Features (BoSUrF) approach are also presented in Section 4.5. The classification of the above-mentioned feature extraction techniques is achieved using the Support Vector Machines (SVM), the K-Nearest Neighbours (K-NN) and the Naive Bayes classifiers. The results obtained are then empirically compared in Section 4.6 to select the best combination of the feature extractor and classifier for automatic detection and classification of thermite weld defects. Section 4.7 concludes the chapter. All experiments were conducted on a 64-bit MSI machine powered by the Nvidia GeForce graphics card with 24 cores, and 32 GB RAM. The source codes were implemented using the Matlab R2019b software under the school license.

4.2 Dataset Description

The dataset used to conduct the experiments was collected from the Transnet Freight Rail (TFR) welding department. A total of 300 thermite weld images representing four classes, namely defect-less, wormholes, shrinkage cavities and inclusions defects were collected, and each class consisted of 75 images. A 5 fold cross-validation method was used for each experiment, where, in each fold (model), 240 images (60 per class) were used for training, and 60 images (15 per class) were used for

validation purposes. Figures 4.1 to 4.4 depict some of the sample images from each class.



Figure 4.1: Defect-less



Figure 4.2: Wormholes



Figure 4.3: Shrinkage cavities



Figure 4.4: Inclusions

4.3 Image Enhancement

The collected thermite weld images are characterised by low contrast and narrow pixel dynamic range values. Thus, image enhancement techniques were used to improve image quality and increase defect visibility. The enhancement technique used

in this work is the Contrast Limited Histogram Equalisation (CLAHE) technique, and it was applied on every weld joint image using the steps explained by Algorithm 1. As mentioned in Section 3.2, the CLAHE technique overcomes the noise enhancement artefact introduced by the Adaptive Histogram Equalisation (AHE) technique by clipping the histogram before using the Cumulative Distribution Function (CDF) as the transform function.

4.3.1 Parameter Evaluation

The clip factor parameter, ρ in Equation 3.5, controls the enhancement rate of the output image. Figure 4.5 shows the original image and the effect of ρ on the output image. It can be observed that there is no significant difference in the image quality between the output image and the original image at a small clip factor value (see Figure 4.5b). Then, at a higher clip factor value, defect visibility (dark regions) in an image is significantly increased; however, there is an introduction of noise on the produced image (see Figure 4.5e to Figure 4.5f). Therefore, an optimal clip factor is found at $\rho=0.01$. As shown in Figure 4.5c, defect visibility in an output image is increased compared to the original image, but the over enhancement and over-amplification of noise is avoided.

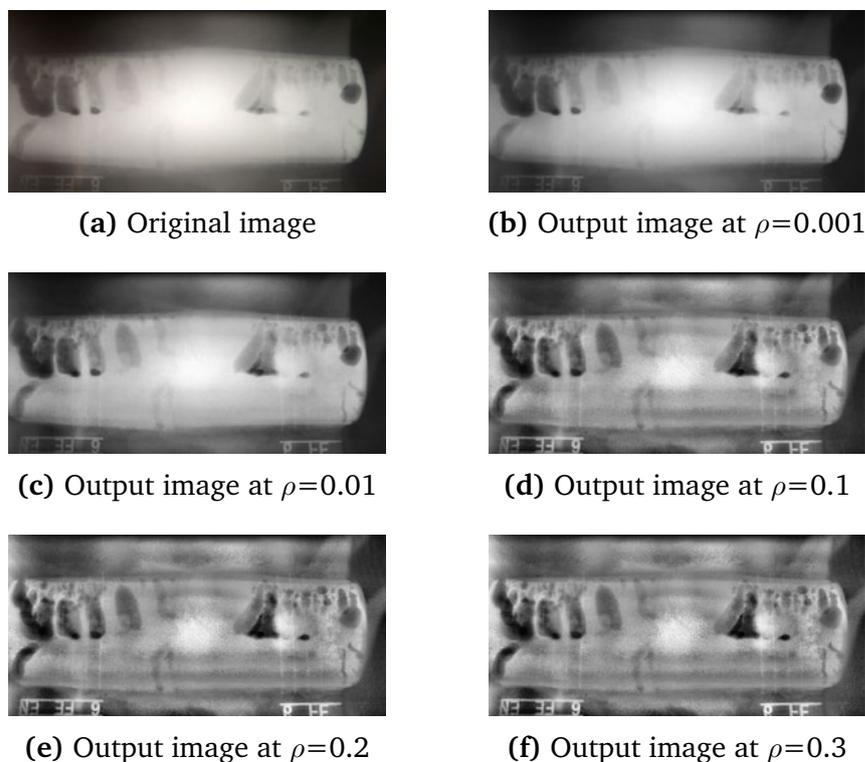


Figure 4.5: Image Enhancement using CLAHE at varying clip factor values

4.4 Weld Joint Extraction

After image enhancement, the next step is to extract the weld joint as the Region of Interest (RoI) from the thermite weld image background. The segmentation method used in this work is the Chan-Vese ACM which segments image regions without the use of edge information. The extraction of the weld joint was carried out using Algorithm 2 detailed in Section 3.3. The Chan-Vese ACM was applied on each image in the dataset to obtain the contour at the boundaries of the weld joint (see Figure 4.6a). The obtained contour was then segmented to obtain the size and coordinates of the weld joint (see Figure 4.6b). The coordinates were then superimposed on the original image, and the rectangular bounding box was placed on the boundaries of the segmented region (see Figure 4.6c). The area inside the bounding box was then cropped, and the cropped image represents the weld joint as the RoI (see Figure 4.6d).

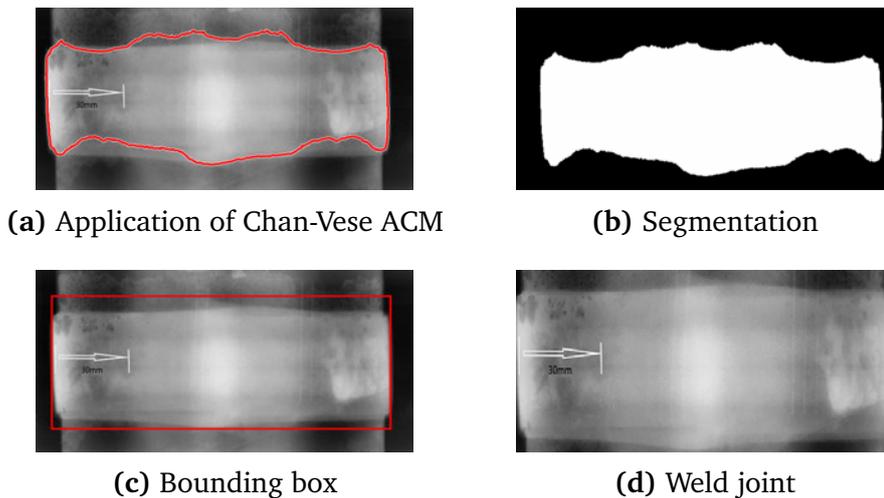


Figure 4.6: Weld joint RoI steps

Figures 4.7 to 4.10 depict the weld joint RoI extraction for sample images in each class. It can be observed in Figure 4.8 that some images that contain wormholes defects, needed to be post-processed in order to achieve accurate segmentation and weld joint extraction. This is understandable since wormholes defects are characterised by multiple “worm-like” dark patterns introduced by gas entrapment during the thermite welding process. On the contrary, shrinkage cavities and inclusion defects were easily segmented as they are mostly characterised by a single shape representing a defect. Shrinkage cavities usually appear as a straight line (see Figure 4.9), and they are caused by the poor pre-heating temperature of rail ends during the thermite welding. In comparison, inclusions are irregular in shape (see Figure

4.10) and are caused by the presence of foreign objects. The post-processing technique employed in this work to remove residual spots on images segmented by the proposed Chan-Vese ACM is based on dilation discussed in Section 3.3.4. Figure 4.11 shows the segmentation accuracy of the proposed Chan-Vese ACM on the thermite weld images per each class. The image is considered to be successfully segmented if there is no dark spots on the segmented weld joint RoI after post-processing. The proposed method achieved the segmentation accuracy of 100% on images that belong to defect-less and shrinkage cavities classes. Furthermore, the segmentation accuracy of 97% was achieved on images belonging to the inclusions class, the lowest segmentation accuracy of 83% was achieved on the images belonging to the wormholes class. On average, the proposed Chan-Vese ACM achieved the segmentation accuracy of 95%.

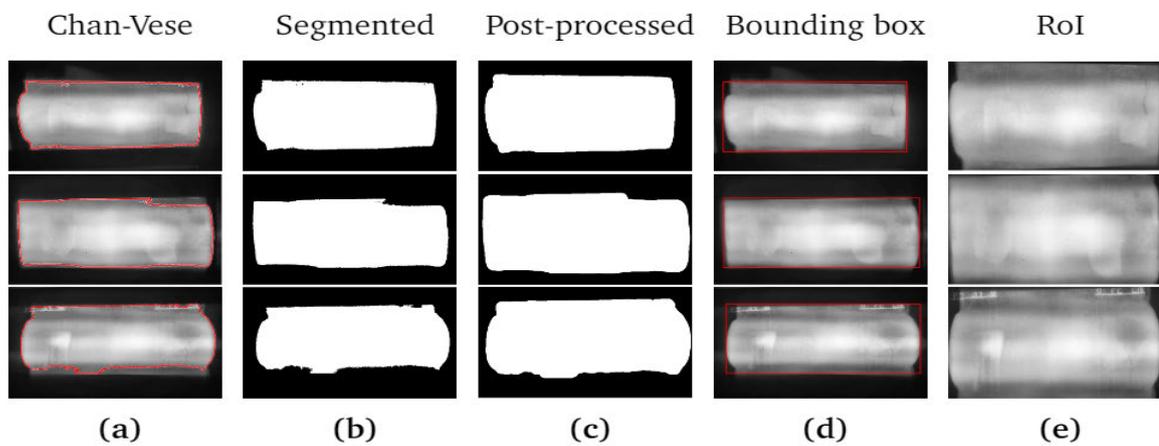


Figure 4.7: Weld joint extraction in a defect-less class

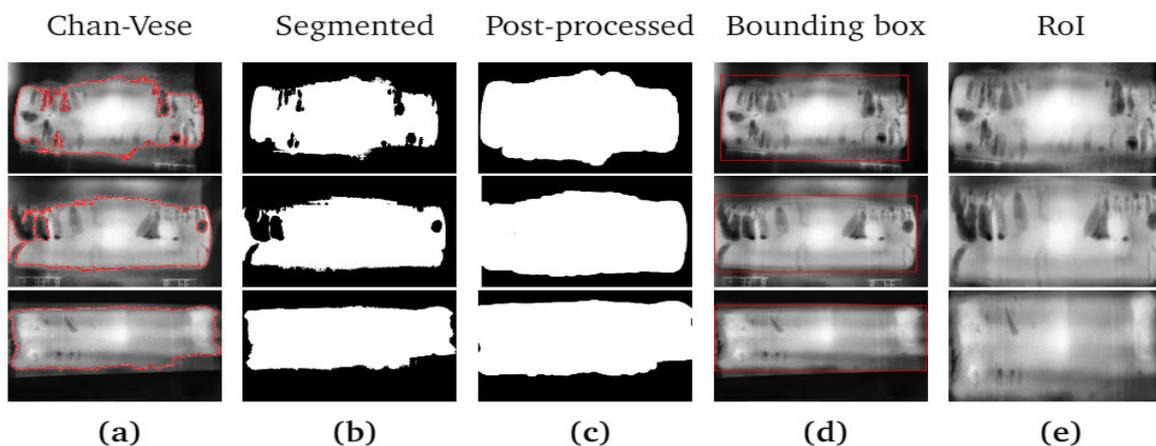


Figure 4.8: Weld joint extraction in Wormholes class

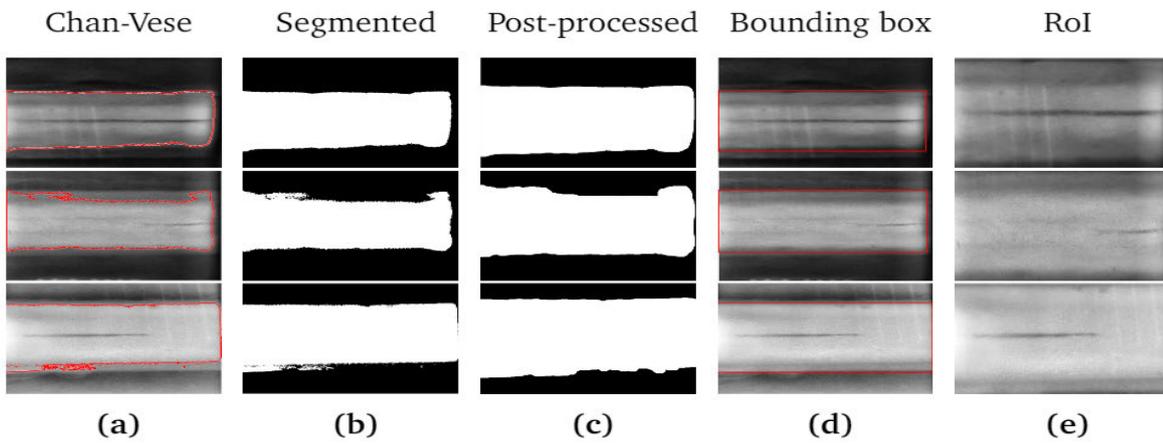


Figure 4.9: Weld joint extraction shrinkage cavities class

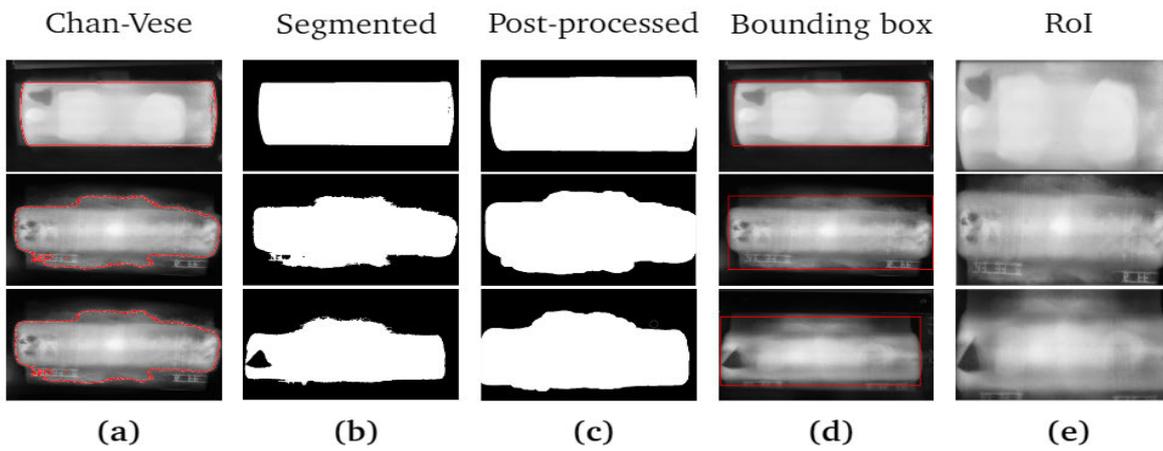


Figure 4.10: Weld joint extraction in Inclusions class

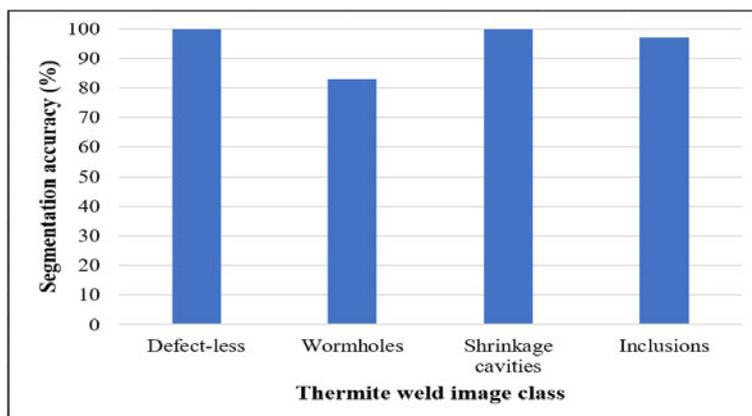


Figure 4.11: Segmentation accuracy for each class

4.5 Feature Classification

After extracting the weld joint as the ROI from the image background, two feature extraction methods namely, the LBP descriptor and BoSURF approach were applied to represent every weld joint image in the dataset as a feature vector. Inputs to the feature extraction algorithms are weld joint images of size 300×700 pixels. The performance of each feature extraction method was evaluated using three machine learning classification algorithms, namely K-NN, SVM and Naïve Bayes. The 5-fold cross-validation method was employed to train and validate each classification algorithm, where in each model, 240 feature vectors (60 per class) were used to train the classifiers, and 60 feature vectors (15 per class) were used to validate the classifiers. The following subsections detail the process.

4.5.1 Classification of the Local Binary Patterns Features

Features extraction using the LBP descriptor was performed on every weld joint image in the dataset using Algorithm 3 presented in Section 3.4. To investigate the impact of the LBP cell size parameter on the classification accuracy, features were extracted at increasing cell size parameters, namely: [6 14], [12 28], [30 70] and [60 140]. Additionally, the number of neighbours and radius parameters were kept at a combination of (8,1). As outlined in Section 3.4, this combination is ideal since uniform patterns are found to occur more at 8 neighbouring pixels. Furthermore, the long feature vector length is avoided at this combination.

Classification using the K-Nearest Neighbours

Feature classification using the K-NN classifier was achieved using Algorithm 7 presented in the previous chapter. As mentioned in Section 3.5, the value of the K parameter in the K-NN classifier can have a significant impact on the classification accuracy. It controls the number of training feature vectors considered when assigning a class label to the unknown feature vector. A smaller K value would mean the classifier is sensitive to the outliers, while a higher value would mean the neighbourhood includes too many vectors from other classes. Therefore, different values of K, ranging from 1 to 7 ($K = 1, 3, 5, 7$) were experimented in this work at every LBP cell size parameter to obtain the highest classification results at each cell size. Tables 4.1 to 4.4 shows the best confusion matrix results obtained by the K-NN classifier at optimal but different value of K in each LBP cell size parameter. The results are based on the 5 fold cross-validation method, where in each model, 240 feature vectors (60 per class) were used to train the K-NN classifier, and 60 feature vectors (15 per class) were used for validation purposes.

Table 4.1: Confusion matrix using LBP and 5-NN at [6 14] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 72 | 1 | 0 | 2 |
| Wormholes | 1 | 72 | 0 | 2 |
| S. Cavities | 5 | 0 | 69 | 1 |
| Inclusions | 3 | 2 | 1 | 69 |

Table 4.2: Confusion matrix using LBP and 5-NN at [12 28] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 70 | 0 | 1 | 4 |
| Wormholes | 1 | 72 | 0 | 2 |
| S. Cavities | 5 | 0 | 69 | 1 |
| Inclusions | 5 | 1 | 2 | 67 |

Table 4.3: Confusion matrix using LBP and 1-NN at [30 70] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 70 | 1 | 0 | 4 |
| Wormholes | 3 | 70 | 0 | 2 |
| S. Cavities | 5 | 0 | 69 | 1 |
| Inclusions | 5 | 3 | 2 | 65 |

Table 4.4: Confusion matrix using LBP and 3-NN at [60 140] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 69 | 0 | 2 | 4 |
| Wormholes | 3 | 70 | 1 | 1 |
| S. Cavities | 4 | 1 | 69 | 1 |
| Inclusions | 3 | 3 | 5 | 64 |

The average classification accuracy achieved by the K-NN classifier (at optimal K value) in each LBP cell size was calculated from the obtained confusion matrix results using Equation 3.62. It was calculated by taking the mean average precision per class. In each class, precision is calculated as the ratio of feature vectors correctly classified to belong to a class (true positive) to the actual number of class feature vectors (true positive and true negative). Figure 4.12 illustrates the highest classification accuracy achieved at optimal K value of the K-NN classifier in each LBP cell size parameter. It can be observed that the highest, overall classification accuracy of 94% was achieved at the optimal K value and LBP cell size parameter of 5 and [6 14] respectively. Additionally, the lowest classification accuracy of 90.67% was achieved at the K value and LBP cell size parameter of 3 and [60 140] respectively. The accuracy slightly decreases with an increase in the cell size parameter. This slight decrease in the accuracy indicates that the K-NN classifier generally provides better classification performance at a smaller spatial scale of the LBP descriptor. Furthermore, it should be noted from the confusion matrix results that the classes which contribute to the decrease in the classification accuracy at increasing cell size are the inclusions and shrinkage cavities. Therefore, the LBP cell size parameter has been found to have an impact on the classification accuracy achieved by the K-NN classifier.

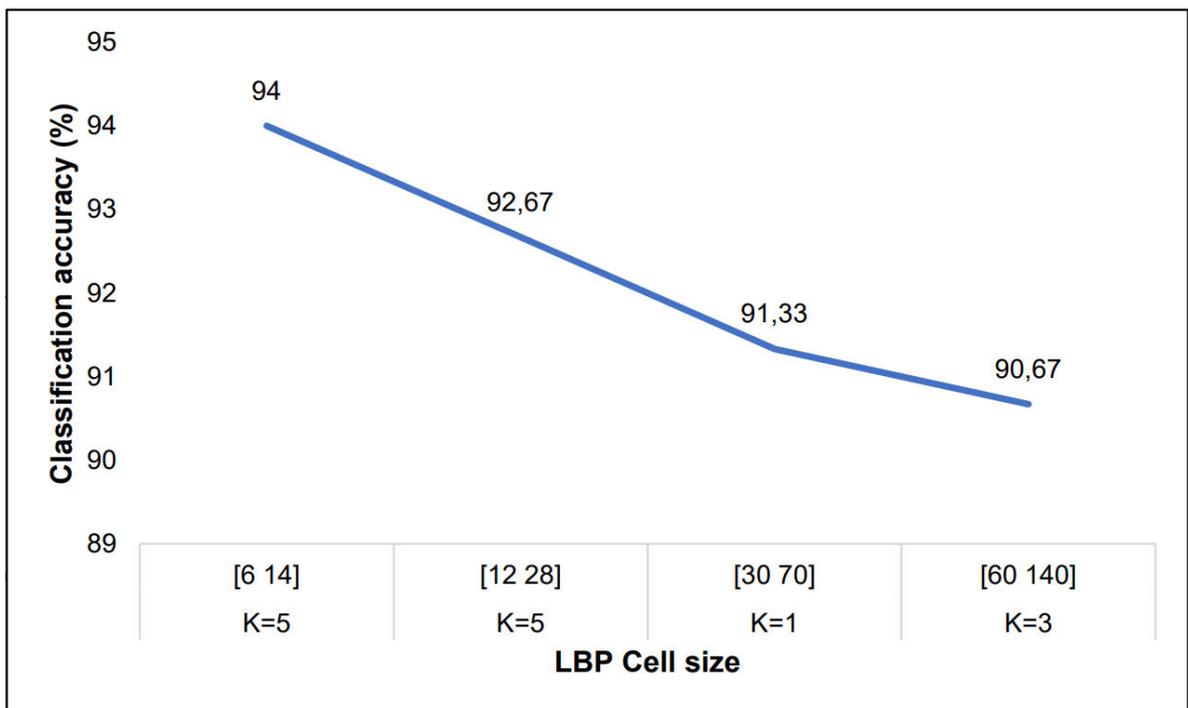


Figure 4.12: Classification accuracy of the K-NN classifier at varying LBP cell size parameter

Classification using the Support Vector Machines

Feature classification using the SVM classifier was achieved using Algorithm 6 presented in the previous chapter. The non-linear SVM with the Radial Basis Function (RBF) kernel was used. As already mentioned in Section 3.5, the kernel width σ can significantly impact the classification results. The σ parameter in the RBF kernel determines the reach of a single training feature vector. A very high σ value would mean the training feature vectors have a far reach, in contrary, a very low σ value would mean the training feature vectors have a closer reach. This means that higher σ values will yield a decision boundary that depends on the closest feature vectors, ignoring feature vectors further away. Subsequently, the lower values of σ will yield a decision boundary that considers feature vectors that are furthest from the decision boundary. Thus, to prevent the formation of a highly flexed decision boundary and the decision boundary that is linear, different σ values ranging from 2^{-4} to 2^4 ($\sigma = 2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^1, 2^2, 2^3, 2^4$) were experimented to obtain the optimal value of σ at each LBP cell size parameter. Tables 4.5 to 4.8 show the best confusion matrix results obtained by the SVM classifier at optimal but different σ value in each LBP cell size based on the 5 fold cross-validation method.

Table 4.5: Confusion matrix using LBP and SVM ($\sigma = 4$) at [6 14] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 69 | 4 | 0 | 2 |
| Wormholes | 2 | 70 | 3 | 0 |
| S. Cavities | 1 | 2 | 72 | 0 |
| Inclusions | 2 | 1 | 2 | 70 |

Table 4.6: Confusion matrix using LBP and SVM ($\sigma=0.25$) at [12 28] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 67 | 4 | 0 | 4 |
| Wormholes | 5 | 66 | 4 | 0 |
| S. Cavities | 0 | 2 | 73 | 0 |
| Inclusions | 3 | 0 | 0 | 72 |

Table 4.7: Confusion matrix using LBP and SVM ($\sigma=0.5$) at [30 70] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 67 | 5 | 0 | 3 |
| Wormholes | 7 | 68 | 0 | 0 |
| S. Cavities | 0 | 5 | 69 | 1 |
| Inclusions | 2 | 0 | 0 | 73 |

Table 4.8: Confusion matrix using LBP and SVM ($\sigma=0.5$) at [60 140] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 68 | 2 | 2 | 3 |
| Wormholes | 7 | 68 | 0 | 0 |
| S. Cavities | 0 | 6 | 69 | 0 |
| Inclusions | 3 | 0 | 0 | 72 |

The average classification accuracy achieved by the SVM classifier (at optimal σ value) in each LBP cell size parameter has been obtained, and it was calculated from the confusion matrix results using Equation 3.62. Figure 4.13 depicts the highest classification accuracy achieved by the SVM classifier at optimal σ value in each LBP cell size parameter. The highest, overall classification accuracy achieved by the SVM classifier is 93.33%, and it was achieved at a σ value of 4 and [6 14] LBP cell size parameter. Furthermore, the lowest classification accuracy of 91.67% was achieved at a σ value of 0.5 and LBP cell size parameter of [60 140]. Similar to the results obtained by the K-NN classifier, there is a slight decrease in the classification accuracy obtained by the SVM classifier at increasing LBP cell size parameter. Also similar to the conclusion made on the results obtained by the K-NN classifier; the LBP cell size parameter has been experimentally found to impact the classification results obtained by the SVM classifier. The advantage of a small LBP cell size parameter is that features can be extracted in very small local regions in an image; this makes it possible to detect features that otherwise could not be detected at a larger spatial scale. On the downside, a small LBP cell size yields a longer feature vector, and this greatly increases the computation cost.

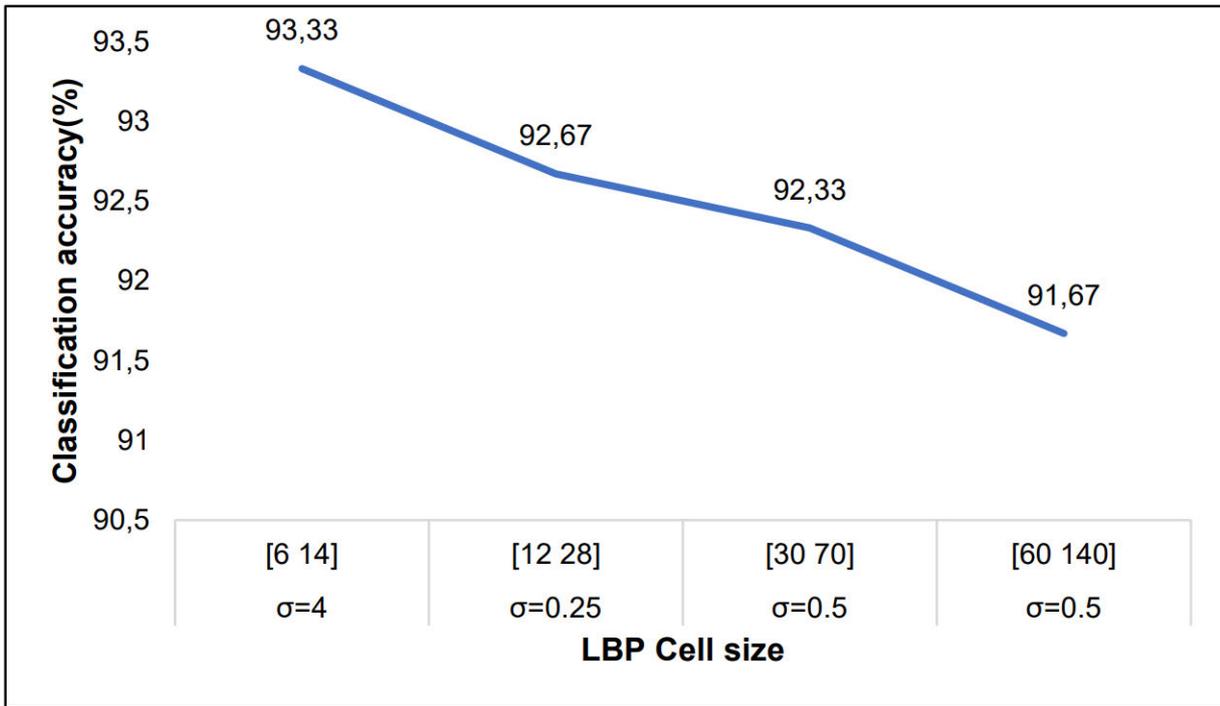


Figure 4.13: Classification accuracy of the SVM classifier at varying LBP cell size parameter

Classification using Naive Bayes

Feature classification using the Naive Bayes classifier was achieved using Algorithm 8. The Naive Bayes classifier is simple, fast and known to perform effectively on a limited dataset. Furthermore, the Naive Bayes classifier requires less parameter tuning than other classifiers such as the SVM and K-NN. Tables 4.9 to 4.12 depict the confusion matrix results obtained by the Naive Bayes classifier at each LBP cell size parameter.

Table 4.9: Confusion matrix using LBP and Naive Bayes at [6 14] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 54 | 11 | 5 | 5 |
| Wormholes | 14 | 59 | 2 | 0 |
| S. Cavities | 8 | 5 | 52 | 10 |
| Inclusions | 7 | 2 | 10 | 56 |

Table 4.10: Confusion matrix using LBP and Naive Bayes at [12 28] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 52 | 13 | 7 | 3 |
| Wormholes | 11 | 59 | 3 | 2 |
| S. Cavities | 1 | 6 | 61 | 7 |
| Inclusions | 0 | 2 | 9 | 64 |

Table 4.11: Confusion matrix using LBP and Naive Bayes at [30 70] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 67 | 3 | 3 | 2 |
| Wormholes | 3 | 71 | 1 | 0 |
| S. Cavities | 5 | 5 | 63 | 2 |
| Inclusions | 2 | 5 | 12 | 56 |

Table 4.12: Confusion matrix using LBP and Naive Bayes at [60 140] cell size

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 60 | 12 | 1 | 2 |
| Wormholes | 11 | 53 | 11 | 0 |
| S. Cavities | 5 | 6 | 58 | 6 |
| Inclusions | 3 | 2 | 5 | 65 |

Figure 4.14 shows the average classification accuracy achieved by the Naive Bayes classifier at varying LBP cell size parameters. The accuracy was calculated from the confusion matrix results using Equation 3.62. Contrary to the classification accuracies obtained by the K-NN and SVM classifiers, the classification accuracy achieved by the Naive Bayes classifier increases with an increase in the cell size parameter. This increase indicates that the Naive Bayes classifier generalises better on the feature vectors extracted at a large LBP spatial scale. However, there is a slight decrease in the classification accuracy after [30 70] cell size. The highest classification accuracy obtained by the Naive Bayes classifier is 85.66%, and it was achieved at [30 70] LBP cell size parameter.

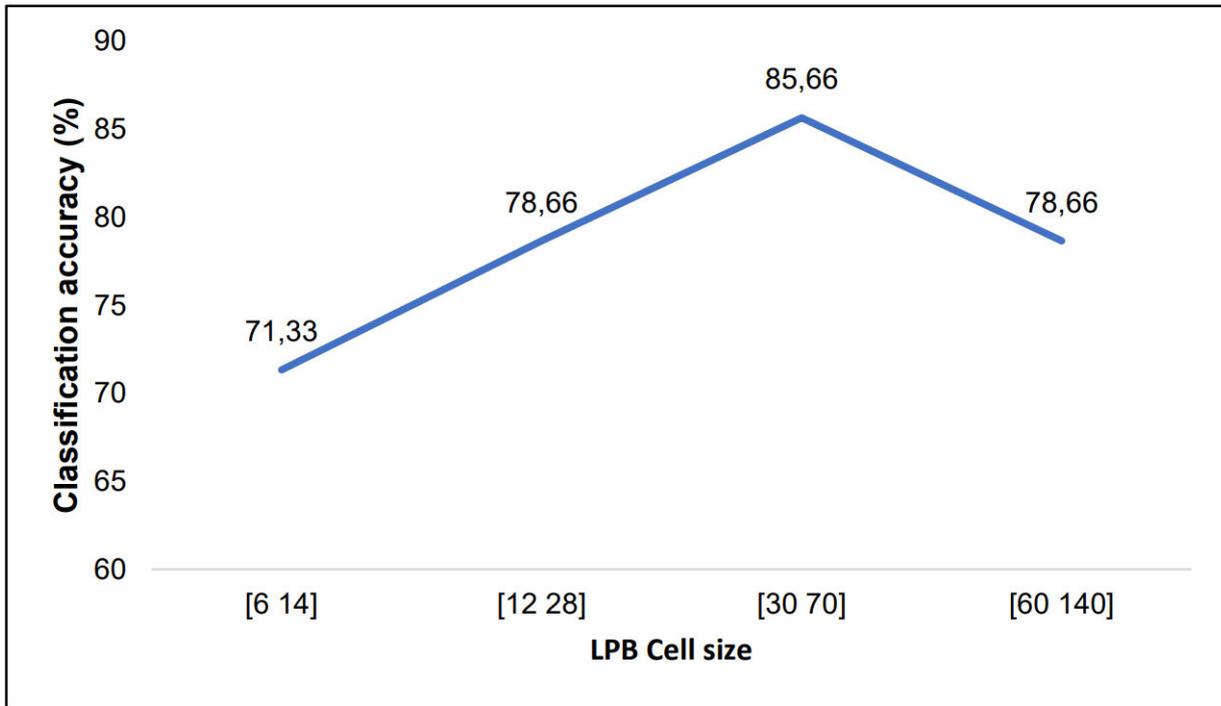


Figure 4.14: Classification accuracy of the Naive Bayes classifier at varying LBP cell size parameter

4.5.2 Best Classifier for LBP Features

Table 4.13 depicts the highest classification accuracy achieved by each classifier on features extracted by the LBP descriptor. It is shown that the best classifier for classifying the LBP features is the K-NN ($K=5$) classifier with the classification accuracy of 94%. The second best classifier is the SVM ($\sigma = 4$) with the classification accuracy of 93.33%. The Naive Bayes classifier achieved the lowest classification accuracy of 85.66%.

Table 4.13: Highest classification accuracy by each classifier for LBP features

| Method | LBP parameter | Classifier type | Accuracy(%) |
|-------------------|--------------------------|---------------------------------|--------------|
| LBP + K-NN | Cell size: [6 14] | K-NN: $K = 5$ | 94.00 |
| LBP + SVM | Cell size: [6 14] | SVM: $\sigma = 4$ | 93.33 |
| LBP + Naive Bayes | Cell size: [30 70] | Naive Bayes | 85.66 |

4.5.3 Classification of the Bag of Speeded Up Robust Features

Keypoint detection and description on the weld joint images was achieved using the SURF descriptor according to the steps listed by Algorithm 4 explained in the previous chapter. As mentioned in Section 3.4, the *minHessian* value used to filter out unstable keypoints was selected to be 500. The BoVW method was applied on the keypoints descriptors to learn a codebook and represent every weld joint image as a global feature vector according to the steps listed in Algorithm 5. For codebook construction; the *K*-means clustering algorithm was applied on the randomly sampled SURF keypoint descriptor vectors, and in an attempt to obtain the optimal codebook size, the codebook was constructed at increasing number of codewords ranging from 200 to 2000.

Classification using the K-Nearest Neighbors

Feature classification using the K-NN classifier was achieved using Algorithm 7 presented in the previous chapter. Different values of K ranging from 1 to 7 (K = 1, 3, 5, 7) were experimented to find the optimal value of K at each codebook size. Tables 4.14 to 4.17 show the best confusion matrix results obtained using the K-NN classifier at optimal but different value of K in each codebook size parameter based on the 5 fold cross-validation method. In each model, 240 feature vectors (60 per class) were used to train the classifier, and 60 feature vectors (15 per class) were used to validate the classifier.

Table 4.14: Confusion matrix using BoSURF and 3-NN at 200 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 63 | 2 | 5 | 5 |
| Wormholes | 1 | 65 | 3 | 6 |
| S. Cavities | 7 | 2 | 65 | 1 |
| Inclusions | 7 | 3 | 2 | 63 |

Table 4.15: Confusion matrix using BoSURF and 5-NN at 800 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 67 | 2 | 3 | 3 |
| Wormholes | 3 | 65 | 4 | 3 |
| S. Cavities | 2 | 1 | 70 | 2 |
| Inclusions | 2 | 2 | 2 | 69 |

Table 4.16: Confusion matrix using BoSURF and 5-NN at 1400 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 67 | 2 | 3 | 3 |
| Wormholes | 1 | 69 | 2 | 3 |
| S. Cavities | 4 | 1 | 68 | 2 |
| Inclusions | 4 | 2 | 4 | 65 |

Table 4.17: Confusion matrix using BoSURF and 3-NN at 2000 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 67 | 3 | 1 | 4 |
| Wormholes | 5 | 65 | 3 | 2 |
| S. Cavities | 2 | 0 | 70 | 3 |
| Inclusions | 3 | 2 | 2 | 68 |

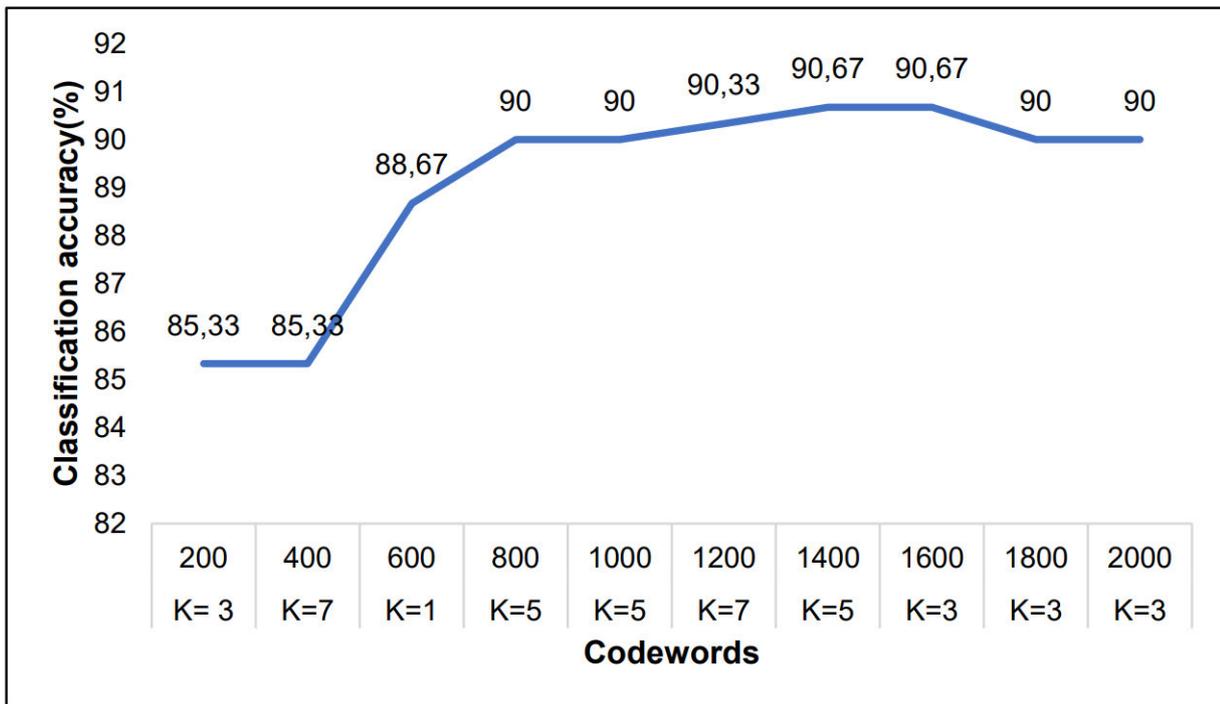
**Figure 4.15:** Classification accuracy of the K-NN classifier at varying codebook size parameter

Figure 4.15 shows the average classification accuracy achieved at optimal K value of the K-NN classifier at increasing number of codewords. The accuracy was calcu-

lated from the confusion matrix results using Equation 3.62. A significant increase in the classification accuracy can be observed for the first 800 codewords. Afterwards, there is a slight and less significant increase in the classification accuracy on the remaining codewords. The highest classification accuracy achieved by the K-NN (K=5) classifier is 90.66%, and it was obtained at 1400 codewords.

Classification using the Support Vector Machines

Feature classification using the SVM classifier was performed using Algorithm 6 presented in the previous chapter. The non-linear SVM with the RBF kernel was used. Different σ values ($\sigma = 2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^1, 2^2, 2^3, 2^4$) were experimented at each codebook size in order to obtain the optimal value of σ . Tables 4.18 to 4.21 show the best confusion matrix results obtained by the SVM classifier at optimal but different σ value for each codebook size parameter based on the 5 fold cross-validation method.

Table 4.18: Confusion matrix using BoSURF and SVM ($\sigma = 0.5$) at 200 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 68 | 2 | 3 | 2 |
| Wormholes | 4 | 62 | 3 | 6 |
| S. Cavities | 3 | 5 | 63 | 4 |
| Inclusions | 4 | 4 | 8 | 59 |

Table 4.19: Confusion matrix using BoSURF and SVM ($\sigma = 4$) at 800 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 71 | 0 | 2 | 2 |
| Wormholes | 2 | 68 | 1 | 4 |
| S. Cavities | 3 | 3 | 66 | 3 |
| Inclusions | 2 | 3 | 4 | 66 |

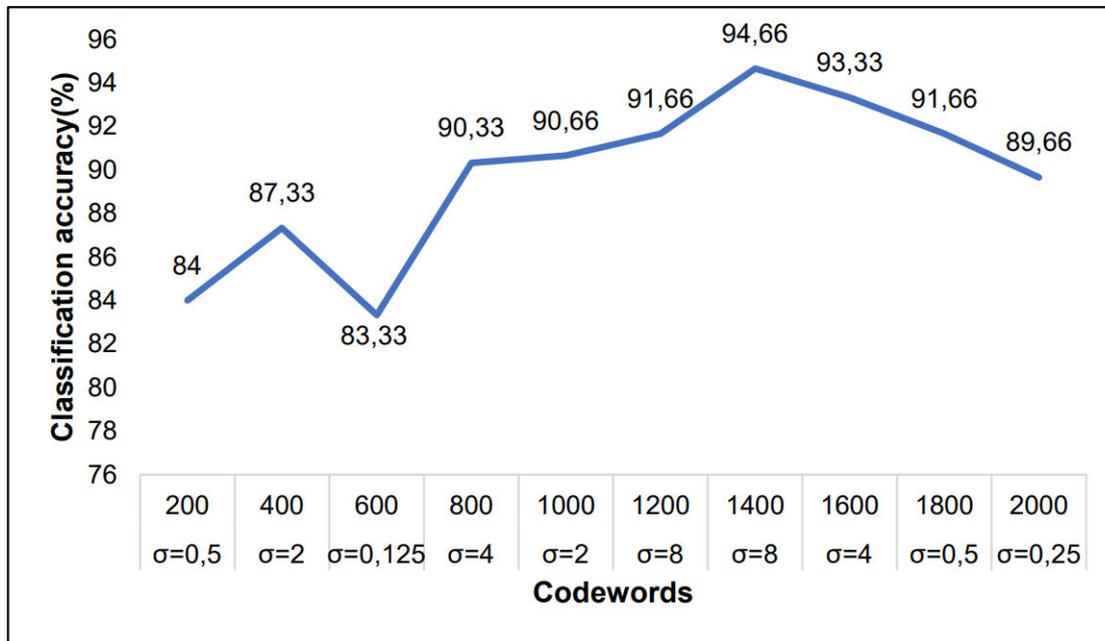
Table 4.20: Confusion matrix using BoSURF and SVM ($\sigma = 8$) at 1400 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 71 | 0 | 2 | 2 |
| Wormholes | 1 | 72 | 1 | 1 |
| S. Cavities | 2 | 0 | 71 | 2 |
| Inclusions | 3 | 0 | 2 | 70 |

Table 4.21: Confusion matrix using BoSURF and SVM ($\sigma = 0.25$) at 2000 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 65 | 2 | 4 | 4 |
| Wormholes | 1 | 68 | 2 | 4 |
| S. Cavities | 0 | 1 | 71 | 3 |
| Inclusions | 4 | 3 | 3 | 65 |

Figure 4.16 shows the average classification accuracy achieved by the SVM classifier (at optimal σ value) in each codebook size parameter. The accuracy was calculated from the obtained confusion matrix results using Equation 3.62. It can be observed that the classification accuracy increases with an increase in the codebook size from 600 to 1400 codewords. After that, there is a linear decrease in the classification accuracy with an increase in the number of codewords. The highest classification accuracy achieved by the SVM classifier (at $\sigma = 8$) for classifying the BoVW features is 94.66%, and it was obtained at the optimal codebook size of 1400 codewords. Subsequently, the lowest classification accuracy achieved by the SVM classifier was obtained at 600 codewords.

**Figure 4.16:** Classification accuracy of the SVM classifier at varying codebook size parameter

Classification using the Naive Bayes Classifier

Feature classification using the Naive Bayes classifier was achieved according to the steps listed by Algorithm 8. Tables 4.22 to 4.25 depict the confusion matrix results obtained by the Naive Bayes classifier at varying LBP cell size parameter.

Table 4.22: Confusion matrix using BoSURF and Naive Bayes at 200 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 51 | 4 | 14 | 6 |
| Wormholes | 6 | 51 | 8 | 10 |
| S. Cavities | 9 | 0 | 61 | 5 |
| Inclusions | 3 | 12 | 8 | 52 |

Table 4.23: Confusion matrix using BoSURF and Naive Bayes at 800 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 55 | 5 | 10 | 5 |
| Wormholes | 7 | 51 | 5 | 12 |
| S. Cavities | 6 | 4 | 62 | 3 |
| Inclusions | 3 | 8 | 12 | 52 |

Table 4.24: Confusion matrix using BoSURF and Naive Bayes at 1400 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 60 | 6 | 7 | 2 |
| Wormholes | 2 | 63 | 5 | 5 |
| S. Cavities | 7 | 2 | 61 | 5 |
| Inclusions | 3 | 8 | 3 | 61 |

Table 4.25: Confusion matrix using BoSURF and Naive Bayes at 2000 codewords

| | Defect-less | Wormholes | S. Cavities | Inclusions |
|-------------|-------------|-----------|-------------|------------|
| Defect-less | 65 | 3 | 4 | 3 |
| Wormholes | 5 | 62 | 7 | 1 |
| S. Cavities | 4 | 2 | 64 | 5 |
| Inclusions | 0 | 8 | 7 | 60 |

Figure 4.17 shows the average classification accuracy achieved by the Naive Bayes classifier at varying codebook size parameter. The accuracy was calculated from the confusion matrix results using Equation 3.62. There is an increase in the classification accuracy for the initial 1200 codewords. After that, the codebook size parameter has a less significant impact on the classification accuracy. The highest classification accuracy achieved by the Naive Bayes classifier is 88.33%, and it was obtained at the optimal codebook size parameter of 1200 codewords.

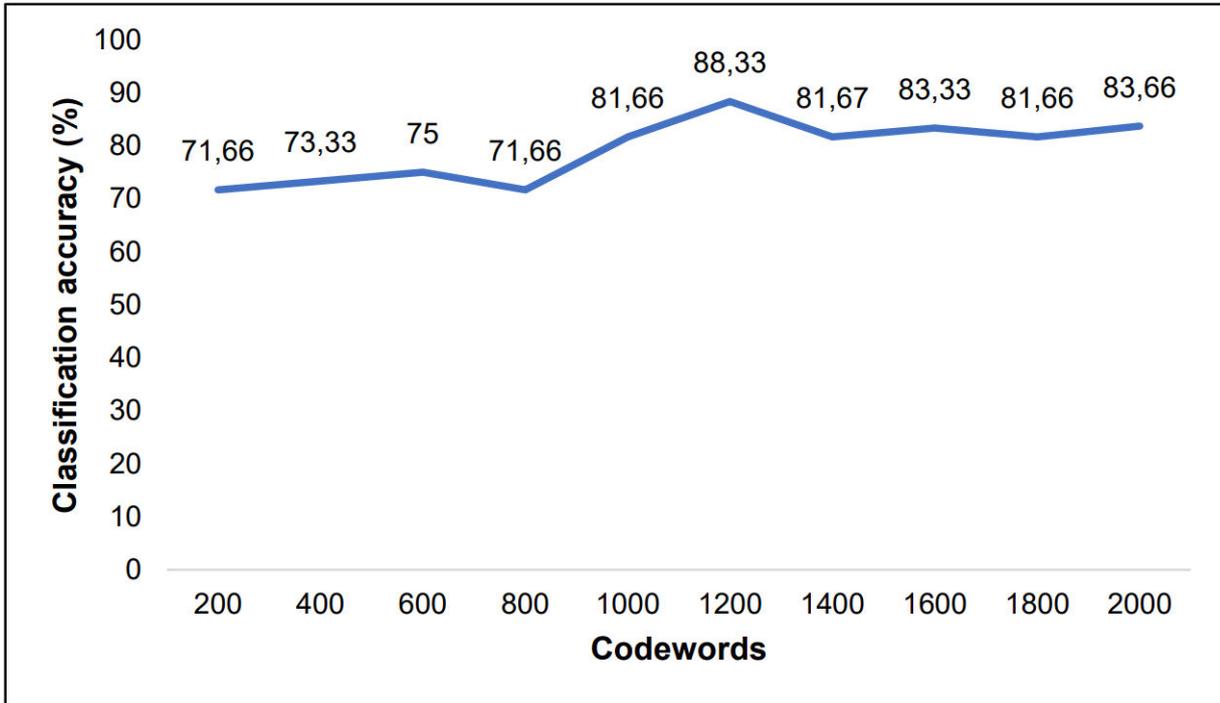


Figure 4.17: Classification accuracy of the Naive Bayes classifier at varying codebook size parameter

4.5.4 Best Classifier for BoSURF Features

Table 4.26 depicts the highest classification accuracy achieved by each classifier on features extracted by the BoSURF approach. It can be observed that the best classifier for classifying the BoSURF features is the SVM (at $\sigma = 8$) classifier with the classification accuracy of 94.66%. The second best classifier is the K-NN (at $K = 5$), with the classification accuracy of 90.66%. Similar to the classification accuracy obtained for classifying LBP features, the Naive Bayes classifier achieved the lowest classification accuracy of 85.66% for classifying the BoSURF features.

Table 4.26: Highest classification accuracy by each classifier for BoSURF features

| Method | parameter | Classifier type | Accuracy(%) |
|----------------------|------------------------|-------------------------------------|--------------|
| BoSURF + K-NN | Codewords: 1400 | K-NN: K = 5 | 90.66 |
| BoSURF + SVM | Codewords: 1400 | SVM: $\sigma = 8$ | 94.66 |
| BoSURF + Naive Bayes | Codewords: 1200 | Naive Bayes | 88.33 |

4.6 Best Method for Detection and Classification of Thermite Weld defects

In Section 4.5, the classification results obtained by each classifier for classifying features extracted by the LBP descriptor and the BoSURF approach were presented. Certain parameters of the feature extraction and classification algorithms were fine-tuned to identify optimal parameter values that achieve the best possible accuracy for classifying thermite weld defects. As shown in Table 4.13, it was found that the K-NN classifier obtains the best classification accuracy for LBP features at the optimal cell size parameter and K value of [6 14] and 5, respectively. Additionally, the best classification accuracy for BoSURF features was obtained by the SVM classifier at the optimal codebook size of 1400 codewords and σ value of 8 (see Table 4.26). The aforementioned best classification accuracies are depicted in Table 4.27, and they are compared to select the best method for automatic detection and classification of thermite weld defects. It can be observed that the best method for detecting and classifying thermite weld defects is achieved by the combination of the BoSURF approach and SVM classifier with the classification accuracy of 94.66%. It should be noted that the 94% classification accuracy achieved by the LBP descriptor and K-NN is not far off; However, since the optimal LBP cell size parameter was found at the small spatial scale, the resulting feature vector length representing each weld joint image is incredibly long (147 500) compared to the feature vector length of only 1400 histograms obtained by the BoSURF approach. Thus, the results clearly indicate that the BoSURF approach outperforms the LBP descriptor in terms of defect detection, classification accuracy and computation cost.

Table 4.27: Highest classification accuracy achieved for LBP and BoSURF

| Method | Optimal parameters | | Feature length | Accuracy(%) |
|---------------------|------------------------|--------------------------------|----------------|--------------|
| LBP + K-NN | Cell size: [6 14] | K = 5 | 147 500 | 94.00 |
| BoSURF + SVM | Codewords: 1400 | $\sigma = 8$ | 1400 | 94.66 |

4.7 Conclusion

This chapter has presented the experimental results of the methods introduced in Chapter 3. The chapter first introduced the dataset used to conduct the experiments. This was followed by applying the CLAHE technique to improve the quality of every image. The weld joint was extracted as the ROI from the background of each enhanced image, where the Chan-Vese ACM was used as a segmentation method. Two feature extraction methods, namely the LBP descriptor and BoSURF approach, were applied on each weld joint image to represent the weld joint as a feature vector. The performance of the feature extraction methods was evaluated using the three classification algorithms, namely the K-NN, SVM and Naive Bayes. Hyperparameter tuning was performed on the feature extraction and classification algorithms to identify the optimal parameters for best classification results. The experimental results indicated that the best method for detecting and classifying thermite weld defects is obtained by combining the BoSURF approach as a feature extractor and the SVM as a classifier.

Chapter 5

Conclusion and Future Work

5.1 Dissertation Conclusion

This work aimed to develop an image processing and machine learning based method to detect and classify thermite weld defects in welded rails automatically. Conventionally, the detection and classification of thermite weld defects in radiography images is conducted manually by a trained RT specialist with loads of work experience. However, questions have been raised regarding the use of human expertise to detect and classify defects. The manual process is deemed to be biased, lengthy and subjective even if the experts conduct it. Therefore, there is a need to develop a method that can automatically detect and classify defects in a robust, fast, reliable and objective manner without human interventions.

An automated thermite weld defect detection and classification method have been developed based on image processing and machine learning techniques. Due to the nature of the obtained thermite weld radiography images, four steps were proposed: image enhancement, image segmentation, feature extraction, and feature classification. The collected images were characterised by poor contrast; therefore, image enhancement techniques were required to improve the image quality and defect visibility. According to the literature study, it was found that the CLAHE technique provides better enhancement results on radiography images compared to other histogram equalisation techniques. Thus, the collected images were enhanced using the CLAHE technique, and image quality was improved.

An algorithm has been developed and applied on the enhanced images to extract the weld joint (RoI) from the image background. The literature study indicated that the edge-based segmentation methods such as Thresholding and Hough transform are effective for a variety of segmentation tasks. However, the collected images contained an irregularly shaped weld joint and a complex image background. This then

made the Thresholding and Hough transform techniques to be ineffective. Therefore, the segmentation of the weld joint was achieved using the Chan-Vese ACM, which is based on the level set methods. Image segmentation using the Chan-Vese ACM allows contours to be represented as a level of a topological function that can merge or split. Thus irregularly shaped image regions can be segmented. The proposed Chan-Vese ACM achieved the segmentation accuracy of 95%

Feature extraction techniques were then applied to the weld joint images to represent every weld joint as a feature vector. The literature categorised these techniques into local and global feature extractors. Local feature extractors were found to have more advantages than global feature extractors as they are invariant to significant image transformations such as rotation, viewpoint and illumination changes. Therefore, two local feature extraction techniques, namely the LBP descriptor and the SURF descriptor, were independently applied on the weld joint images to represent every image as a feature vector.

The SURF descriptor first detects the scale-invariant keypoints before computing a descriptor vector for each keypoint in the image. This meant a single image is represented by many feature vectors for training a classifier; thus, computational costs demand are extensively high. To address this challenge, the BoVW (BoSURF) approach was used to create a codebook in a completely unsupervised learning manner from the unlabelled SURF descriptor vectors. Weld joint images were therefore represented by a single histogram vector that is a count of how many times each codeword appears on the image. The K-means clustering algorithm was used to create visual codewords.

The performance of the two feature extractors was compared using three classifiers, namely the K-NN, SVM and Naive Bayes. These three classifiers were selected due to their effectiveness in modelling a small dataset. Some parameters of the feature extractors and classifiers were fine-tuned to evaluate their impact on the classification performance and to select the best classification results. For feature extractors, these parameters were the LBP cell size and the codebook size on the LBP descriptor and BoSURF approach, respectively. For classifiers, the parameters were the K value and the σ value on the K-NN and SVM classifiers, respectively.

The experiments were conducted based on the 5-fold cross-validation method. It was found that the best method for detecting and classifying thermite weld defects is achieved when BoSURF features are combined with SVM classifier; this combination achieved the classification accuracy of 94.66% at optimal codebook size and σ parameters of 1400 and 8, respectively. To the best of the author's knowledge, lim-

ited research work exist in the literature for the specific objective of detecting and classifying thermite weld defects in welded rails using image processing and machine learning techniques. Thus, the results obtained in this work can be used as a baseline for further research studies and improvement to the topic at hand.

5.2 Recommendation for Future work

To address the limitations of this study (see Section 1.5), the following recommendations should be considered for future work.

1. Image dataset of other thermite weld defects types should be collected to implement a robust method that can detect and classify any thermite weld defects.
2. More thermite weld image dataset should be collected and made publicly available to compare the methods proposed in this work to some of the state of the art methods based on Deep learning approaches.

Bibliography

- [1] R.Ndlela, "Xrs-4 portable x-ray machine operation," *Transnet Ltd, Johannesburg*, 2019.
- [2] O. Phiri, T. Raseala, P. Gibbon, and A. Sibanyoni, "Aluminothermic welding management using lean six sigma methodology on the south african coal export line," *Transnet Ltd, Johannesburg*, 2019.
- [3] G. Yang, D. Li, G. Ru, J. Cao, and W. Jin, "Body height estimation system based on binocular vision," *International Journal of Online Engineering (iJOE)*, vol. 14, p. 177, 04 2018.
- [4] X. Wang, B. Zhou, J. Ji, and B. Pu, "Recognition and distance estimation of an irregular object in package sorting line based on monocular vision," *International Journal of Advanced Robotic Systems*, vol. 16, p. 172988141982721, 02 2019.
- [5] B. Suvdaa, J. Ahn, and J. Ko, "Steel surface defects detection and classification using sift and voting strategy," 2012.
- [6] K. Selvi and D. JohnAravindar, "An industrial inspection approach for weld defects using machine learning algorithm," 2019.
- [7] K. Huang, H. Yang, I. King, and M. Lyu, *Local Learning vs. Global Learning: An Introduction to Maxi-Min Margin Machine*, 11, vol. 117, pp. 625–626.
- [8] Y. Min, B. Xiao, J. Dang, B. Yue, and T. Cheng, "Real time detection system for rail surface defects based on machine vision," *EURASIP Journal on Image and Video Processing*, vol. 2018, p. 3, 12 2018.
- [9] Y. Chen, F. Lawrence, C. Barkan, and J. Dantzig, "Heat transfer modelling of rail thermite welding," *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, vol. 220, pp. 207 – 217, 2006.
- [10] S. Singh. and R. Clark, "Ultrasonic examination of thermite welds in rails based on the arema guidelines," *Materials Evaluation*, vol. 62, pp. 143–148, 02 2004.

- [11] X. Gibert, V. M. Patel, and R. Chellappa, "Deep multitask learning for railway track inspection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 1, pp. 153–164, 2017.
- [12] A. James, W. Jie, Y. Xulei, Y. Chenghao, N. B. Ngan, L. Yuxin, S. Yi, V. Chandrasekhar, and Z. Zeng, "Tracknet - a deep learning based fault detection for railway track inspection," in *2018 International Conference on Intelligent Rail Transportation (ICIRT)*, 2018, pp. 1–5.
- [13] E. Deutschl, C. Gasser, A. Niel, and J. Werschonig, "Defect detection on rail surfaces by a vision based system," in *IEEE Intelligent Vehicles Symposium, 2004*, 2004, pp. 507–511.
- [14] C. Tastimur, M. Karakose, A. Erhan, and A. Ilhan, "Rail defect detection with real time image processing technique," 07 2016, pp. 411–415.
- [15] M. Troscino, A. Shaw, and J. Cunningham, *Automated Track Inspection Vehicle and Method*. 16, 2000.
- [16] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 881–892, 2002.
- [17] A. Wosiak, A. Zamecznik, and K. Niewiadomska-Jarosik, "Supervised and unsupervised machine learning for improved identification of intrauterine growth restriction types," in *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2016, pp. 323–329.
- [18] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18–28, 1998.
- [19] P. Cunningham and S. Delany, "k-nearest neighbour classifiers," *Mult Classif Syst*, 04 2007.
- [20] J. Ham, Yangchi Chen, M. M. Crawford, and J. Ghosh, "Investigation of the random forest framework for classification of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 492–501, 2005.
- [21] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–44, 05 2015.

- [22] M. Rajagopal, M. Balasubramanian, and S. Palanivel, "An efficient framework to detect cracks in rail tracks using neural network classifier," *Computacion y Sistemas*, vol. 22, pp. 943–952, 09 2018.
- [23] B. Yue, Y. Wang, Y. Min, Z. Zhang, W. Wang, and J. Yong, "Rail surface defect recognition method based on adaboost multi-classifier combination," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2019, pp. 391–396.
- [24] Y. Santur, M. Karaköse, and E. Akin, "Random forest based diagnosis approach for rail fault inspection in railways," in *2016 National Conference on Electrical, Electronics and Biomedical Engineering (ELECO)*, 2016, pp. 745–750.
- [25] B. Xiaio, Y. Min, and H. Ma, "Detection of rail fastener based on wavelet decomposition and pca," in *2017 2nd International Conference on Information Technology and Management Engineering (ITME 2017)*, 2017, pp. 163–168.
- [26] X. Gibert, V. M. Patel, and R. Chellappa, "Robust fastener detection for autonomous visual railway track inspection," in *2015 IEEE Winter Conference on Applications of Computer Vision*, 2015, pp. 694–701.
- [27] S. Gao, T. Szugs, and R. Ahlbrink, "Use of combined railway inspection data sources for characterization of rolling contact fatigue," 06 2018.
- [28] Y. Jiang, H. Wang, G. Tian, Q. Yi, J. Zhao, and K. Zhen, "Fast classification for rail defect depths using a hybrid intelligent method," *Optik*, vol. 180, pp. 455–468, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0030402618317984>
- [29] H. Zhang, X. Jin, Q. M. J. Wu, Y. Wang, Z. He, and Y. Yang, "Automatic visual detection system of railway surface defects with curvature filter and improved gaussian mixture model," *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 7, pp. 1593–1608, 2018.
- [30] K. G. Mercy and S. K. Srinivasa Rao, "A framework for rail surface defect prediction using machine learning algorithms," in *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*, 2018, pp. 972–977.
- [31] A. Núñez, A. Jamshidi, and H. Wang, "Pareto-based maintenance decisions for regional railways with uncertain weld conditions using the hilbert spectrum of axle box acceleration," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1496–1507, 2019.
- [32] N. Yao, Y. Jia, and K. Tao, "Rail weld defect prediction and related condition-based maintenance," *IEEE Access*, vol. 8, pp. 103 746–103 758, 2020.

- [33] D. Soukup and R. Huber-Mörk, “Convolutional neural networks for steel surface defect detection from photometric stereo images,” 12 2014.
- [34] L. Shang, Q. Yang, J. Wang, S. Li, and W. Lei, “Detection of rail surface defects based on cnn image recognition and classification,” in *2018 20th International Conference on Advanced Communication Technology (ICACT)*, 2018, pp. 45–51.
- [35] S. Faghieh-Roohi, S. Hajizadeh, A. Núñez, R. Babuska, and B. De Schutter, “Deep convolutional neural networks for detection of rail surface defects,” in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 2584–2589.
- [36] A. Jamshidi, S. Hajizadeh, Z. Su, M. Naeimi, A. Núñez, R. Dollevoet, B. De Schutter, and Z. Li, “A decision support approach for condition-based maintenance of rails based on big data analysis,” *Transportation Research Part C: Emerging Technologies*, vol. 95, pp. 185–206, 2018.
- [37] S. Yanan, Z. Hui, L. Li, and Z. Hang, “Rail surface defect detection method based on yolov3 deep learning networks,” in *2018 Chinese Automation Congress (CAC)*, 2018, pp. 1563–1568.
- [38] Q. Xu, Q. Zhao, G. Yu, L. Wang, and T. Shen, “Rail defect detection method based on recurrent neural network,” in *2020 39th Chinese Control Conference (CCC)*, 2020, pp. 6486–6490.
- [39] X. Song, K. Chen, and Z. Cao, “Resnet-based image classification of railway shelling defect,” in *2020 39th Chinese Control Conference (CCC)*, 2020, pp. 6589–6593.
- [40] A. Sumesh, K. Rameshkumar, K. Mohandas, and R. S. Babu, “Use of machine learning algorithms for weld quality monitoring using acoustic signature,” *Procedia Computer Science*, vol. 50, pp. 316–322, 2015, big Data, Cloud and Computing Challenges.
- [41] S. Seungmin, J. Chengnan, Y. Jiyoung, and R. Sehun, “Real-time detection of weld defects for automated welding process base on deep neural network,” *Metals*, vol. 10, p. 389, 03 2020.
- [42] K. Roumen, K. Roumiana, and V. Todorov, “Defects detection in x-ray images and photos,” 02 2011.
- [43] M. Saja and H. Suhaila, “Image enhancement process on digital radiographic image with weld discontinuities,” *Journal of Mechanical Engineering*, vol. 5, pp. 275–292, 03 2018.

- [44] M. Rajab, T. Elbenawy, and M. W. Alhazmi, "Application of frequency domain processing to x-ray radiographic images of welding defects," *Journal of X-ray Science and Technology*, vol. 15, pp. 147–156, 2007.
- [45] N. Nacereddine, M. Zelmat., S. Belaïfa, and M. Tridi, "Weld defect detection in industrial radiography based digital image processing," *Proceeding of World Academy of Science, Engineering and Technology*, vol. 2, 01 2015.
- [46] M. El-Tokhy and I. Mahmoud, "Classification of welding flaws in gamma radiography images based on multi-scale wavelet packet feature extraction using support vector machine," *Journal of Nondestructive Evaluation*, vol. 34, 09 2015.
- [47] Z. Abidin, M. Anompa, and M. Muhtadan, "Development of welding defects identifier application on radiographic film using gray level co-occurrence matrix and backpropagation," *AIP Conference Proceedings*, vol. 1555, pp. 70–74, 09 2013.
- [48] O. Zahran, H. Kasban, M. El-Kordy, and F. A. El-Samie, "Automatic weld defect identification from radiographic images," *NDT E International*, vol. 57, pp. 26–35, 2013.
- [49] W. Zhihong and X. Xiaohong, "Study on histogram equalization," in *2011 2nd International Symposium on Intelligence Information Processing and Trusted Computing*, 2011, pp. 177–179.
- [50] S. Attia, "Enhancement of chest x-ray images for diagnosis purposes," *Journal of Natural Sciences Research*, vol. 6, pp. 43–47, 2016.
- [51] C. Dang, J. Gao, Z. Wang, F. Chen, and Y. Xiao, "Multi-step radiographic image enhancement conforming to weld defect segmentation," *IET Image Process.*, vol. 9, pp. 943–950, 2015.
- [52] K. Koonsanit, S. Thongvigitmanee, N. Pongnapang, and P. Thajchayapong, "Image enhancement on digital x-ray images using n-clahe," in *2017 10th Biomedical Engineering International Conference (BMEiCON)*, 2017, pp. 1–4.
- [53] Y. Zhang, l. zhang, B. Dai, B. Chen, and Y. Li, "Welding defect detection based on local image enhancement," *IET Image Processing*, vol. 13, 09 2019.
- [54] W. hui Hou, D. Zhang, Y. Wei, J. Guo, and X. Zhang, "Review on computer aided weld defect detection from radiography images," *Applied Sciences*, vol. 10, p. 1878, 2020.

- [55] M. Carrasco and D. Mery, "Segmentation of welding defects using a robust algorithm," 2004.
- [56] F. Mirzaei, M. Faridafshin, A. Movafeghi, and R. Faghihi, "Automated defect detection of weldments and castings using canny , sobel and gaussian filter edge detectors : A comparison study," 2017.
- [57] V. Mutneja, "Methods of image edge detection: A review," *Journal of Electrical Electronic Systems*, vol. 04, 01 2015.
- [58] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 04, pp. 321–331, 01 1988.
- [59] N. Nacereddine, M. Zelmat, S. S. Belaïfa, and M. Tridi, "Weld defect detection in industrial radiography based digital image processing," *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering*, vol. 1, pp. 433–436, 2007.
- [60] M. H. Chowdhury and W. D. Little, "Image thresholding techniques," in *IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing. Proceedings*, 1995, pp. 585–589.
- [61] L. Mahmoudi and A. El Zaart, "A survey of entropy image thresholding techniques," in *2012 2nd International Conference on Advances in Computational Tools for Engineering Applications (ACTEA)*, 2012, pp. 204–209.
- [62] X. Fu-song, "Survey over image thresholding techniques based on entropy," in *2014 International Conference on Information Science, Electronics and Electrical Engineering*, vol. 2, 2014, pp. 1330–1334.
- [63] D. Changying, G. Jianmin, Z. Wang, X. Yulin, and Z. Yalin, "A novel method for detecting weld defects accurately and reliably in radiographic images," *Insight - Non-Destructive Testing and Condition Monitoring*, vol. 58, pp. 28–34, 01 2016.
- [64] V. Sridevi, G. Jianmin, and A. Nirmala, "Inspection of welding images using image segmentation techniques," *International Journal of Engineering Research Technology (IJERT)*, vol. 2, pp. 28–34, 03 2013.
- [65] W. Al-Hameed, Y. Mayali, and P. Picton, "Segmentation of radiographic images of weld defects," *Journal of Global Research in Computer Science*, vol. 4, pp. 28–34, 03 2013.

- [66] A. Mahmoudi and F. Rezagui, "Welding defect detection by segmentation of radiographic images," in *2009 WRI World Congress on Computer Science and Information Engineering*, vol. 7, 2009, pp. 111–115.
- [67] T. Tong, Y. Cai, and D. Sun, "Defects detection of weld image based on mathematical morphology and thresholding segmentation," in *2012 8th International Conference on Wireless Communications, Networking and Mobile Computing*, 2012, pp. 1–4.
- [68] Lu Yu, Qiao Wang, Lenan Wu, and Jun Xie, "Mumford-shah model with fast algorithm on lattice," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 2, 2006, pp. II–II.
- [69] T. F. Chan and L. A. Vese, "A level set algorithm for minimizing the mumford-shah functional in image processing," in *Proceedings IEEE Workshop on Variational and Level Set Methods in Computer Vision*, 2001, pp. 161–168.
- [70] M. B. Gharsallah and E. B. Braiek, "Weld inspection based on radiography image segmentation with level set active contour guided off-center saliency map," *Advances in Materials Science and Engineering*, vol. 2015, pp. 1–10, 2015.
- [71] Y. Boutiche, N. Ramou, and M. B. Gharsallah, "An implicit region-based deformable model with local segmentation applied to weld defects extraction," *World Academy of Science, Engineering and Technology, International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, vol. 6, pp. 1267–1271, 2012.
- [72] F. Mekhalfa and N. Nacereddine, "Multiclass classification of weld defects in radiographic images based on support vector machines," in *2014 Tenth International Conference on Signal-Image Technology and Internet-Based Systems*, 2014, pp. 1–6.
- [73] I. Valavanis and D. Kosmopoulos, "Multiclass defect detection and classification in weld radiographic images using geometric and texture features," *Expert Systems with Applications*, vol. 37, no. 12, pp. 7606–7614, 2010.
- [74] J. Shao, H. Shi, D. Du, L. Wang, and H. Cao, "Automatic weld defect detection in real-time x-ray images based on support vector machine," in *2011 4th International Congress on Image and Signal Processing*, vol. 4, 2011, pp. 1842–1846.
- [75] J. Hassan, A. M. Awan, and A. Jalil, "Welding defect detection and classification using geometric features," in *2012 10th International Conference on Frontiers of Information Technology*, 2012, pp. 139–144.

- [76] R. da Silva, P. Calôba, H. Siqueira, and M. Rebello, "Pattern recognition of weld defects detected by radiographic test," *NDT E International*, vol. 37, no. 6, pp. 461–470, 2004.
- [77] R. Hernández, J. Zapata Pérez, and R. Merino, "Classification of welding defects in radiographic images using an adaptive-network-based fuzzy system," 05 2011, pp. 205–214.
- [78] J. Zapata, R. Vilar, and R. Ruiz, "An adaptive-network-based fuzzy inference system for classification of welding defects," *NDT E International*, vol. 43, no. 3, pp. 191–199, 2010.
- [79] A. S. Ibrahim, A. E. Youssef, and A. L. Abbott, "Global vs. local features for gender identification using arabic and english handwriting," in *2014 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 2014, pp. 000 155–000 160.
- [80] C. Shu, X. Ding, and C. Fang, "Histogram of the oriented gradient for face recognition," *Tsinghua Science and Technology*, vol. 16, no. 2, pp. 216–224, 2011.
- [81] H. Yang and X. A. Wang, "Cascade face detection based on histograms of oriented gradients and support vector machine," in *2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, 2015, pp. 766–770.
- [82] T. I. Dhamecha, P. Sharma, R. Singh, and M. Vatsa, "On effectiveness of histogram of oriented gradient features for visible to near infrared face matching," in *2014 22nd International Conference on Pattern Recognition*, 2014, pp. 1788–1793.
- [83] E. Gao, Q. Gao, and J. Chen, "The welding region extraction technology based on hog and svm," 01 2017.
- [84] F. Liu, Y. Liu, and H. Sang, "Multi-classifier decision-level fusion classification of workpiece surface defects based on a convolutional neural network," *Symmetry*, vol. 12, p. 867, 05 2020.
- [85] A. A. Moghaddam and L. Rangarajan, "Classification of welding defects in radiographic images," *Pattern Recognit. Image Anal.*, vol. 26, no. 1, p. 54–60, Jan. 2016.
- [86] D. Mery and C. Arteta, "Automatic defect recognition in x-ray testing using computer vision," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017, pp. 1026–1035.

- [87] D. Mery and C. Arteta, "Automatic defect recognition in x-ray testing using computer vision," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017, pp. 1026–1035.
- [88] Y. Kai, D. Qian, T. Sun, M. Zhang, and S. Zhang, "Weld defect detection based on completed local ternary patterns," 12 2017, pp. 6–14.
- [89] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," vol. 3951, 07 2006, pp. 404–417.
- [90] N. Kong, "A literature review on histogram equalization and its variations for digital image enhancement," *International journal of innovation, management and technology*, 2013.
- [91] K. Kitti, T. Saowapak, P. Napapong, and T. Pairash, "Image enhancement on digital x-ray images using n-clahe," 08 2017, pp. 1–4.
- [92] M. Rahmani, M. Idris, R. Ramli, and H. Arof, "Segmentation of crescent shape blood cells using circular hough transform," *Indian Journal of Science and Technology*, vol. 8, 06 2015.
- [93] X. Chen, B. M. Williams, S. R. Vallabhaneni, G. Czanner, R. Williams, and Y. Zheng, "Learning active contour models for medical image segmentation," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 11 624–11 632.
- [94] Y. Tian, M.-q. Zhou, Z.-k. Wu, and X.-c. Wang, "A region-based active contour model for image segmentation," in *2009 International Conference on Computational Intelligence and Security*, vol. 1, 2009, pp. 376–380.
- [95] L. Cai and Y. Wang, "A phase-based active contour model for segmentation of breast ultrasound images," in *2013 6th International Conference on Biomedical Engineering and Informatics*, 2013, pp. 91–95.
- [96] B. Lucia, *Genetic snakes: Active contour models by genetic algorithms*, 01 2007, vol. 8, pp. 177–194.
- [97] N. Salman, B. Ghafourand, and G. Hadi, "Medical image segmentation based on edge detection techniques," *Advances in Image and Video Processing*, vol. 3, 04 2015.
- [98] M. Ben Gharsallah and E. Ben Braiek, "Image segmentation for defect detection based on level set active contour combined with saliency map," in *2015 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, 2015, pp. 388–392.

- [99] P. Bumrungrun, "Defect detection in textile fabrics with snake active contour and support vector machines," *Journal of Physics: Conference Series*, vol. 1195, p. 012006, 04 2019.
- [100] M. Gharsallah and E. Braiek, "Weld inspection based on radiography image segmentation with level set active contour guided off-center saliency map," *Advances in Materials Science and Engineering*, vol. 2015, pp. 1–10, 12 2015.
- [101] A. Kaushik, C. Prakashand, and S. Mathpal, "Edge detection and level set active contour model for the segmentation of cavity present in dental x-ray images," *International Journal of Computer Applications*, vol. 96, 06 2014.
- [102] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations," *Journal of Computational Physics*, vol. 79, no. 1, pp. 12–49, 1988.
- [103] D. Mumford and J. Shah, "Optimal approximation by piecewise smooth function and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, 07 1989.
- [104] H. Jiang-Hua, Q. Lei, and F. Tian-Qi, "Image processing based on mathematical morphology in camouflage," in *2013 Seventh International Conference on Image and Graphics*, 2013, pp. 269–272.
- [105] D. Chudasama, T. Patel, S. Joshi, and G. Prajapati, "Image segmentation using morphological operations," *International Journal of Computer Applications*, vol. 117, pp. 16–19, 05 2015.
- [106] K. C. Tatikonda, C. M. Bhuma, and S. K. Samayamantula, "The analysis of digital mammograms using hog and glcm features," in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2018, pp. 1–7.
- [107] M. Hassaballah, A. Ali, and H. Alshazly, *Image Features Detection, Description and Matching*, 02 2016, vol. 630, pp. 11–45.
- [108] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognit.*, vol. 29, pp. 51–59, 1996.
- [109] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

- [110] S. Caroline, B. Thierry, and C. Frelicot, "An extended center-symmetric local binary pattern for background modeling and subtraction in videos," vol. 1, 03 2015.
- [111] K. Pavel and L. Ladislav, *Novel Texture Descriptor Family for Face Recognition*, 05 2019, pp. 37–47.
- [112] H. Sebastian and U. Andreas, "A scale- and orientation-adaptive extension of local binary patterns for texture classification," *Pattern Recognition*, vol. 25, 03 2015.
- [113] D. Agnew, "Efficient use of the hessian matrix for circuit optimization," *IEEE Transactions on Circuits and Systems*, vol. 25, no. 8, pp. 600–608, 1978.
- [114] D. Bradley and G. Roth, "Adaptive thresholding using the integral image," *J. Graphics Tools*, vol. 12, pp. 13–21, 01 2007.
- [115] E. Oyallon and J. Rabin, "An analysis of the surf method," *Image Processing On Line*, vol. 5, pp. 176–218, 07 2015.
- [116] P. Pui and J. Minoi, *Keypoint Descriptors in SIFT and SURF for Face Feature Extractions*, 02 2018, pp. 64–73.
- [117] M. Hanmandlu, H. Choud, and J. Dash, "Detection of fabric defects using fuzzy decision tree," *International Journal of Signal and Imaging Systems Engineering*, vol. 9, p. 184, 01 2016.
- [118] R. Entezari-Maleki, A. Rezaei, and B. Minaei, "Comparison of classification methods based on the type of attributes and sample size," *JCIT*, vol. 4, pp. 94–102, 09 2009.
- [119] F. Bouillot, P. Poncelet, and M. Roche, "Classification of small datasets: Why using class-based weighting measures?" 06 2014.
- [120] A. Mahdavi-Shahri, J. Karimian, A. Javadi, and M. Houshmand, "Multi-label classification of small samples using an ensemble technique," 12 2017.
- [121] I. Shadeed, J. Alwan, and D. Abd, "The effect of gamma value on support vector machine performance with different kernels," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, p. 5497, 10 2020.
- [122] D. Nurwaha, "Comparison of kernel functions of support vector machines: A case study for the solar cell output power prediction," *International Journal of Energy Applications and Technologies*, vol. 7, pp. 1–6, 04 2020.

- [123] M. Hussain, S. K. Wajid, A. Elzaart, and M. Berbar, “A comparison of svm kernel functions for breast cancer detection,” in *2011 Eighth International Conference Computer Graphics, Imaging and Visualization*, 2011, pp. 145–150.
- [124] R. Timofte, T. Tuytelaars, and L. Van Gool, “Naive bayes image classification: Beyond nearest neighbors,” vol. 7724, 11 2012, pp. 689–703.