
# Machine Learning Models for Recognizing Curve-shaped Abnormalities in Different Image Modalities

By

**Fakher Mohammad**

A Thesis Submitted for the Degree of Doctor of Philosophy in Computer Science at the University of Buckingham

October 2022

# DECLARATION

I hereby declare that the presented thesis submitted to the University of Buckingham has not been previously published or written by another person, except where due references have been made. I also declare that the thesis has not been previously submitted for any qualification degree at the University of Buckingham or any other university.

***Fakher Mohammad***

# ABSTRACT

This thesis is concerned with using machine learning algorithms for the analysis of different image modalities for the presence of abnormal features or shapes. This challenge appears in various crucial computer vision applications in science, engineering, medicine, and art. Different image modalities abnormal features/regions are often very specific to the applications, the capturing tools, and the subjects of the scenes that were captured. The appearance of certain types of feature abnormalities often indicates potentially serious faulty defects in the imaged objects. We only consider two applications: Inspecting cracks in building materials such as glass façades and concrete surfaces using digital camera images and determining irregularity properties of tumour lesion borders from ultrasound (US) scan images. In the first case, abnormalities appear as cracks that could endanger life and infrastructure. At the same time, irregularity of the tumour border, which is highly correlated to the malignancy of the tumour, could reduce the patient's recovery chances if not recognised/treated early.

Image abnormalities could be manifested by unexpected textural changes in the region of interest or by shape changes of the object(s) of interest. In the case of glass/concrete cracks, human inspectors can easily look out for visible discontinuities in image intensities that are not related to shadows or reflections, but there is a shortage of experienced inspectors for high-rise buildings. Recognising tumour border irregularity in US images conducted by expert sonographers, with many years of clinical training, by observing the overall contour shape of the lesion besides taking into account changes to tissue texture profile surrounding the border. In this thesis, we exploit advances in machine learning to develop AI algorithms that help trigger timely actions to remedy building facades faults and support diagnostic decisions for surgery or treatment of cancer patients. We mostly adopted *handcrafted feature* algorithms but conducted limited experiments on deep learning CNN models, if only to test their viability. The black-box nature of CNN models makes them less attractive for clinical use.

The main difficulties faced, at least initially, that required improvisation includes (1) very limited work in the literature that dealt with a reasonably similar task to those investigated in this thesis, (2) the non-existence of sufficiently large datasets readily useable for developing our AI schemes, and (3) the lack of clear commonly agreed meaning/descriptors of normality/abnormalities of image objects. For each of these items, the degree of difficulty is not the same in the 2 cases. Furthermore, what works for the building façade problem may not be applicable to the other. Besides differences in image modalities, for the building façade case, many suspect objects may be found in a single image, while the second case requires segmenting a single tumour object.

Glass datasets were formed using commercially available drones and collected through Google search. Suspect crack segments were extracted using a state-of-the-art edge detector, and for the development of AI methods, we defined somewhat innovative feature vectors representation of otherwise well-

understood concepts of linearity, curvature, or connected pixel configuration information. For an alternative approach, common texture features such as Uniform Local Binary Pattern (ULBP) and Histogram of Oriented Gradients (HOG) were extracted and analysed in addition to testing the viability of using CNN models. These methods achieved accuracies ranging from 70% using the Histogram of Linearity (HOL) on the low-resolution glass crack dataset and 99% using CNN models on the concrete crack dataset. ULBP achieved 95% on the high-resolution dataset when images were partitioned, and HOG achieved the highest accuracy of 98% on the concrete dataset. A prototype tool for crack recognition from pre-recorded videos demonstrated the efficacy of the handcrafted feature models.

For nodule border irregularity, two datasets of thyroid cancer US images are collected from a hospital in China. We avoided the burden of accurate manual or automatic border segmentation by estimating the border via bi-cubic spline interpolation from a relatively small amount of Region of Interest (ROI) points marked by radiologists. Again, two approaches for designing the sought-after AI schemes are adopted: (1) using geometric features representation of lesion interpolated border morphology, and (2) texture analysis on image data in ribbons of different sizes around the interpolated borderline. For the first approach, several AI methods were developed using distance functions between border and normal reference shapes (e.g. ellipse, gaussian, and convex hull fitted/built from the interpolated border points, as well as convex hull corners fitted ellipse). The distances functions were analysed in the spatial/frequency domain or by Topological Data Analysis (TDA). An innovative method inspired by Fractal Dimensions (FD) was designed that measures border perimeter at different scales. The second approach involved texture analysis of LBP, HOG, and HOL features that were extracted from sectorized border ribbons. We also tested the viability of using two CNN architectures in transfer learning modes. Unlike the first case study of building façade abnormality analysis, the methods based on the morphological features of the interpolated border performed significantly better than texture analysis in border ribbons, with FD inspired method achieving the highest accuracy of 86% on internal testing, followed by distances from the fitted ellipse from convex hull corners with an accuracy of 88% on external testing. Among the texture analysis-based methods, the sector-wise ribbon-based ULBP performed the best, with an accuracy of 72% and 77% on internal and external testing, followed by HOG, while HOL did not give satisfactory results. The deep learning model achieved higher accuracy of 89% on internal testing but a lower accuracy on external testing of 84%. Finally, various methods from both approaches are combined using decision/score level fusion schemes and a decision tree (DT). Most schemes improved the performances of different significance, with score level fusion giving the best overall accuracy of 90%, 92%, and 96% on internal, external and external testing with agreed class labels between different radiologists.

***I dedicate this thesis to my wife and my three children for supporting me and always being patient during my PhD.***

# ACKNOWLEDGMENT

I would like to express my deepest gratitude to my first supervisor Professor Sabah Jassim, for his support and patience throughout this PhD. I also would like to thank my second supervisor Dr Alaa AlZoubi for his help and support. Also, a special thank you to the TenD lab coordinator Mr Hongbo Du for always being helpful throughout the PhD project. Last but not least, I want to thank TenD AI Medical Technologies Ltd., the project's sponsor, for providing the necessary data and funding for the study.

# Table of Contents

# LIST OF FIGURES

# LIST OF TABLES

# ABBREVIATIONS

| | |
|---|---|
| 3PixelConf | 3-connected pixel configurations |
| Acc | Accuracy |
| AI | Artificial Intelligence |
| ALL | Glass crack datasets of the mixture of D4K, DHD, and DG datasets |
| CAD | Computer Aided Diagnosis |
| CNN | Convolutional Neural Networks |
| D4K | Glass crack dataset of 4K image quality |
| DG | Glass crack dataset collected from Google search |
| DHD | Glass crack dataset of HD image quality |
| DL | Deep Learning |
| DT | Decision Tree |
| ED | Edge Drawing |
| Equi-angdisp | interpolated border points sampling method based on equal angular displacements. |
| Equi-arclength | interpolated border points sampling method based on equal arc length |
| FD | Fractal Dimensions |
| FFT | Fast Fourier Transform |
| HOC | Histogram of Curvature |
| HOG | Histogram of oriented gradients |
| HOL | Histograms of Linearity |
| Irreg | Accuracy of irregular lesion border (sensitivity) |
| kNN | k-Nearest Neighbour |
| LBP | Local Binary Patterns |
| ML | Machine Learning |
| MRI | Magnetic Resonance Imaging |
| Reg | Accuracy of regular lesion border (specificity) |
| ROI | Region of Interest |

| | |
|---|---|
| SVD | Singular Value Decomposition method |
| SVM | Support Vector Machines |
| TDA | Topological Data Analysis |
| UAV | Unmanned Aerial Vehicle |
| ULBP | Uniform Local Binary Patterns |
| US | Ultrasound |
| σ | Standard Deviation |

# Chapter 1: Introduction

Recent rapid advances in machine Learning (ML) and artificial intelligence (AI) coupled with a reasonably affordable variety of imaging sensors that could be deployed onboard fixed or mobile robotic devices have resulted in deploying of a growing number of automatic image processing and analysis applications in different areas such as inspection, security, and medical diagnostics. The more success is achieved in such applications, the more opportunities and challenges emerge for automatic image analysis tasks carried out manually by highly trained experts, perhaps conducted in risky environments or inaccessible sites. For example, new opportunities in relation to the deployment of image processing/analysis in hazardous environments such as fires and disaster incidents are a growing area of research interest. Similarly, due to a shortage of expert clinicians/and radiologists, urgent needs have emerged to use AI algorithms to analyse various medical images to support diagnostics decision-making. Such image analysis tasks involve detecting or recognizing certain characteristics of regions or objects of interest in images or real-time recorded videos. This thesis is concerned with automating the recognition of abnormal shapes that could be associated with faults/defects/diseases of the imaged object(s) of interest. These types of tasks include but are not limited to the recognition of broken glass in the façade of high-rising buildings, dangerous cracks in concrete slaps of buildings or bridges, cracks/damages in solar cells, or irregularities of tumour mass boundary correlated with cancer malignancy. The time complexity, cost, and personal risk factors of such tasks, besides shortages in highly skilled workers and/or highly trained specialized radiologists, are the main incentive for developing AI algorithms for such challenging image analysis schemes.

Thus the thesis aims to automate the recognition of abnormal shapes in images in three applications: recognition of broken glass in the façade of high-rising buildings, cracks in the concrete surfaces of buildings, and irregularities of tumour mass boundaries. This chapter presents the problem domain and motivations and identifies the research gap. Then, it presents this study's aim, objectives, and contributions. Finally, an overview of the thesis structure is presented.

## 1.1 Problem Statement

Abnormal defects that appear on the surface of different building materials, such as glass façades, concrete, metal plates etc., can be captured by photography with digital cameras and used as indicators of deterioration of material quality or defects. On the other hand, detecting abnormalities (or variations) in medical images of body tissues/organs (e.g., ultrasound (US) tumour scans) also provides helpful indicators of the presence/absence of different diseases or state of disease progression. In all cases, abnormal artefacts are manifested as abnormalities in the texture patterns of the corresponding image modalities. Hence, the problem is a typical case of image texture analysis in the search for deviations from specific expected structures. For detecting cracks in a glass panel, the analysis must distinguish

non-crack image shapes related to naturally appearing reflections/shadows of nearby buildings/trees. To detect abnormality in the border of scanned tumours, we need to be able to translate the established medical knowledge used in distinguishing irregular borders from regular ones relevant to the examined disease. Texture abnormalities/variations in images from different sources are important characteristics for object recognition and classification. Consequently, our approach will examine existing knowledge of image texture features and, if necessary, modify and/or develop new innovative abnormal texture feature indicators. The essential requirement for investigating its relevance to our tasks should be amenable to AI analysis, i.e., it must have some reasonable discriminating powers that can be determined automatically.

This thesis investigates and analyses textures that appear in different image modalities. We shall focus on typical application-oriented case studies: (1) natural images of building material in the search for faulty material defects and (2) medical US tumour scan images to assess lesion border. Several AI-based methods for both building material and medical US image assessments/classifications are proposed, and their performances are compared.

This thesis's first case study components are designed to assess RGB images of glass panels and concrete surfaces for possible cracks, especially in highly risky sites. Automatically recognizing these cracks results in a safe marking of locations of material defects and can save construction companies a lot of time and money. For glass crack recognition, we aim mainly to develop an automatic crack recognition method for the assessment of the glass façade of high buildings such as skyscrapers (see Figure.1.1). Such systems are essential since any damage to the building's outer decorations could put the pedestrians on the streets below into a risk. Furthermore, manual inspections of the glass façade of such high buildings are time-consuming and cost a lot of money besides putting risks on the workers. The biggest challenge for developing such automatic systems is, though, the appearance of other objects on the surface of the glass panels beside the actual cracks. These objects can easily confuse any possible computer vision-based methods for recognizing cracks and other damages. The appearance of these objects on the glass surface is caused by a high number of reflections coming from objects such as trees, clouds, and opposite buildings. Other artefacts, such as the appearance of objects inside the building through the transparent glass panels or dirt and weathering, can also be found on the building's façade. Furthermore, some of the building's façade glass have carvings and can appear as different textures on the RGB images and therefore be confused with the actual cracks. All these artefacts can affect the image qualities and reduce the performance of any automatic crack recognition methods.

(a) High building with a glass façade [1].

(b) Shanghai tower [2].

Figure 1.1: High building glass façade.

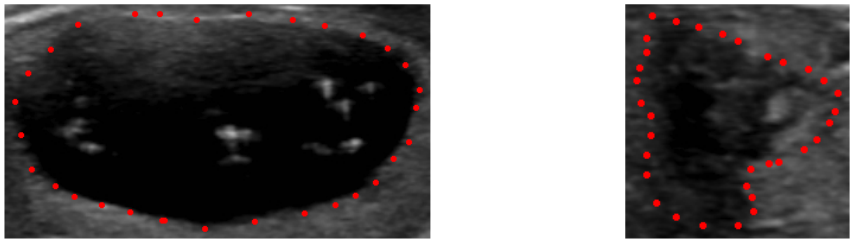
In contrast to glass panels, there are no reflections on concrete surfaces, but many types of artefacts, such as dirt, holes, paints, and uneven surfaces beside the cracks. The cracks on the concrete differ in nature from those on the glass surfaces. While the number of glass cracks tends to be high, and in some types of glasses all over the place, there are often few cracks, and most of the time, only one massive crack on the concrete surfaces (see Figure 1.2). A further challenge facing computer vision-based solutions for such crack recognition tasks is the variations in image illuminations caused by taking the photos at different daytimes and weathers.

Figure 1.2: Glass and concrete cracks [3], [4].

To the best of our knowledge, no works have been reported for the recognition of the building's glass façade cracks. Few works on glass crack assessment in production have been reported; however, these images are taken under ideal illumination where no reflections are visible. On the contrary, many works are reported for recognising defects, including cracks in other building materials such as concrete, asphalt, and mortar. The current work will investigate a generic approach using traditional ML and deep learning methods for crack recognition and evaluate them on glass façades and concrete surfaces.

The second case study of this work covers investigating several techniques relating to certain aspects of cancer characteristics recognition. More specifically, we investigate border irregularities of the thyroid cancer nodules using different texture and shape analysis methods. The irregularity of the border of the cancer lesions is one of the signs contributing to the doctor's decision on the malignancy of the

nodule. Many screening modalities, such as Magnetic Resonance Imaging (MRI), Computerized Tomography CT scan, X-ray, Ultrasound (US) etc., exist for cancer diagnostics. MRI, CT-scan, and X-ray all use electromagnetic waves, which may affect the body if it is exposed to radiation for so long or so often. In contrast, US imaging systems use harmless sound frequencies reflected from the tissues inside the body to build real-time images. They are also more affordable and simpler to operate on, making them popular in numerous medical examinations, including the diagnosis of cancer. However, due to the distinctive nature of the US images, their analysis is quite challenging for AI-based methods in opposite to natural image modalities. There is a perception that ultrasound images are of poor quality even though comparing US image quality may not follow the usual human vision characterization of natural images.



(a) (b)

Figure 1.3: (a) Regular and (b) Irregular thyroid cancer nodule borders. (Red dots are ROI points).

Determining the border of a tumour is ideally done by an appropriately reliable tumour segmentation method which is a very challenging task. Therefore, it would be helpful if medical experts (or other trained persons) could highlight the region of interest by marking a finite number of Region of Interest (ROI) points on the tumour border (see Figure. 1.3). However, how accurate is the marked ROI points depends on the radiologist's experience, and it is a subjective process. Thus, ML-based methods can support radiologists and increase efficiency in cancer diagnosis. Another challenge is the limited availability of sufficiently large medical datasets. There are very few publicly available datasets; obtaining new datasets is challenging due to patients' privacy concerns and the difficulties in obtaining their consent. Another more serious challenge is the difficulty of obtaining the "standard ground truth" for the samples in the datasets that trained medical experts can agree on. In other words, high inter and intra-observer variations in describing the lesion characteristics in US images make the development and assessment of the performance of the AI algorithms challenging.

## 1.2 Aims and Objectives

This research aims to develop Machine Learning (ML) solutions for object abnormality classifications in visual images in two application domains. The first application domain covers the assessment of building materials for damages such as glass and concrete cracks, while the second involves the assessment and analysis of US cancer images for lesion border irregularity recognition. We aim to develop AI methods that utilize existing commonly used features for texture analyses and invent new

innovative features specific to abnormality recognition in the application domain. For crack recognition, we seek certain features that discriminate the crack's characteristics from other objects/structures found on the surface of construction materials. In the case of the cancer signs (nodule border irregularity), we aim to identify discriminative texture features or shape descriptors already used in other application domains or invent new ones for irregularity recognition and assessment. In developing any AI-based method for border irregularity of the cancer lesion, the accurate border must be segmented, or a proper segmentation method needs to be developed, which is not in the scope of this thesis. Therefore, the ML method for border irregularity recognition needs to rely on the rather few ROI points made by experienced doctors to mark the border of the lesion. This will require developing methods for border approximation from ROI points before any border irregularity assessments can be conducted. With these in mind, this work aims toward the following objectives:

- To survey and understand the state-of-the-art geometrical and textural features for object abnormality recognition in general, cracks in building material, and nodule border irregularity in particular.
- To develop generic computer vision-based methods for automatic crack recognitions in both building's glass façade and concrete surfaces.
- To search for publicly available datasets and collect and process new sufficient glass and concrete crack datasets with class label annotations to evaluate the proposed methods for crack recognition.
- To demonstrate the viability of the proposed methods in a software prototype for automatic crack recognition.
- To survey and investigate existing methods for recognising irregularity signs in thyroid cancer from US images and related works.
- To develop methods for borderline approximation and border irregularity recognition.
- To visualize irregular regions on the border of the thyroid cancer lesions.
- To develop evaluation protocols appropriate for evaluating various proposed methods and testing scenarios.
- To investigate the performance of the proposed methods and provide a summary of recommendations and future directions.

## 1.3 Contributions

The research work in this thesis achieved the following contributions, which we divided into two parts as follows :

**Glass and Concrete Crack Recognition:**

1- Two novel methods for crack recognition are proposed based on histograms of linearity (HOL) and Histogram of Curvature indicators (HOC).
2- We developed a block-wise texture feature-based method for crack recognition by adapting the proposed methods of HOL and HOC and the commonly used texture features of HOG and ULBP.
3- Two deep learning-based methods are proposed for glass and concrete crack recognition and compared with traditional ML methods.
4- Three datasets of glass façades were collected using a drone and Google search and cropped and annotated manually.
5- A prototype application for crack recognition is developed to demonstrate the feasibility of the proposed methods in detecting cracks in glass façades.

**Peer-Reviewed Publications:**

1. F. Mohammad, A. AlZoubi, D. Hongbo, and S. Jassim, "Automatic glass crack recognition for high building façade inspection," in *Mobile Multimedia/Image Processing, Security, and Applications 2020*, Online Only, United States, May 2020, p. 32. doi: 10.1117/12.2567409.

2. F. Mohammad, A. AlZoubi, H. Du, and S. Jassim, "A generic approach for automatic crack recognition in buildings glass façade and concrete structures," in *Thirteenth International Conference on Digital Image Processing (ICDIP 2021)*, Singapore, Singapore, Jun. 2021, p. 70. doi: 10.1117/12.2601061.

**Border Irregularity Recognition of Thyroid Cancer Lesion:**

1- A novel method for cancer lesion borderline approximations from the ROI points using cubic spline interpolation and two methods for sampling the new borderline points are proposed.
2- New methods for border irregularity recognition are proposed based on distances of approximated borderline to several reference shapes, such as a fitted ellipse, a Gaussian shape, a convex hull, and a fitted ellipse from the convex hull.
3- Proposed a novel method inspired by Fractal Dimensions (FD) based on the perimeter of the nodule measured at different scales and a fitted ellipse for border irregularity recognition.
4- We developed two novel methods for border irregularity recognition based on border distance analysis using TDA and FFT spectrum number of spikes.
5- A new simple iterative classifier is developed to evaluate single feature-based methods such as FD-inspired and FFT-based methods.

6- Two methods are proposed based on analysing pixel intensities in ribbon sectors around the lesion's border by adopting common LBP and HOG features.
7- Decision and score level fusion and decision tree (DT) based decision mining is proposed to combine decisions of various proposed methods for lesion border irregularity recognition.
8- Several evaluation protocols are developed for testing the proposed methods. The protocols aim to select the best-performing model for external testing.
9- Methods based on Convolutional Neural Networks (CNN) using transfer learning by fine-tuning for thyroid cancer irregularity recognitions are developed and compared with our proposed hand-crafted feature-based methods.

**Peer-Reviewed Publications:**

1. F. Mohammad, A. Alzoubi, H. Du, and S. Jassim, "Machine learning assessment of border irregularity of thyroid nodules from ultrasound images," in *Multimodal Image Exploitation and Learning 2022*, Orlando, United States, May 2022, p. 6. doi: 10.1117/12.2618470.

2. F. Mohammad, A. Alzoubi, H. Du, and S. Jassim, "Irregularity Recognition of Tumor Border in Ultrasound Thyroid Scans Without Segmentation," in Annual Conference on Medical Image Understanding and Analysis MIUA 2022, Cambridge, UK, July 2022 (www.miua2022.com/)

3. Border Irregularity Assessment of Thyroid Cancer Lesion based on Morphological Features. (Journal: *Biomedical Signal Processing and Control BSPC* January 2023)(in preparation)

## 1.4 An Overview of the Research Methodology

Throughout the thesis, we mainly exploit various handcrafted ML methods for abnormality recognition but explored some off-the-shelf deep learning methods based on transfer learning for comparison. The proposed method's performances are analysed and compared, and the findings are summarized. The research approaches for the two case studies for abnormality recognition are slightly different and described separately. In the **first case study,** we collected several datasets of building glass façade cracks and then applied several ML methods exploiting either the texture variations or the geometrical properties of cracked and non-cracked edge segments from images of glass and concrete surfaces. Our method in this part of the thesis consists of four steps (1) pre-processing the input image either by edge detection or image partitioning; (2) extracting either geometrical or textural features; (3) building feature histograms; (4) classification. The steps are illustrated in Figure 1.4.

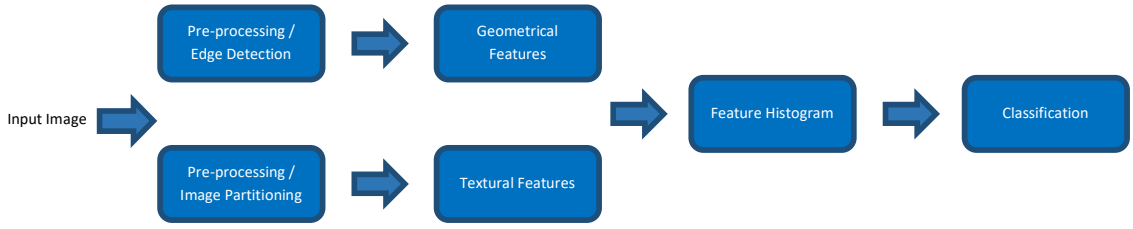



Figure 1.4: First case study methodology

In the **second case study,** two approaches and fusion schemes are followed: (1) border morphological feature analysis based on ROI points, (2) exploiting texture variations in a ribbon around the lesions borderline and (3) exploiting various fusion schemes of the methods from both approaches.

In the first approach, the ROI points are interpolated to approximate the borderline for morphological feature extraction based either on borderline distances or FD-inspired. The distance function is analysed in the FFT domain or by TDA. The second approach involves exploiting the pixel intensity variations around the borderline using a ribbon. The steps for both approaches are illustrated in Figure 1.5 below.

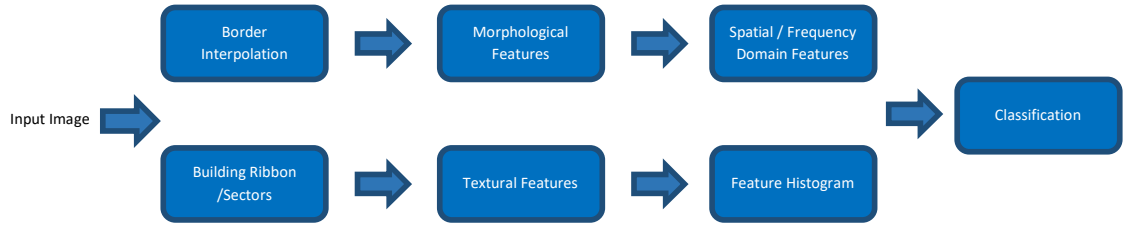



Figure 1.5: Second case study methodology.

Given that we proposed various methods based on different features and approaches for cancer border irregularity recognition, the methods are fused at the decision and score levels and combined using DT.

## 1.5 Structure of the Thesis

The rest of the thesis is organized as follows:

**Chapter 2:** Presents the background of ML and AI algorithms for image analyses and their applications. Here the basics and typical pipelines of both handcrafted features and deep learning-based algorithms are described, and their differences are illustrated. The different assessment algorithms for ML are presented and discussed. The chapter ends by illustrating the main challenges facing any ML algorithm.

**Chapter 3:** In this chapter, the background, theory, and procedures of image texture analysis are introduced, in addition to describing the commonly used texture features in the literature in general for computer vision applications and describing the background of the adopted features used in this thesis

in detail. The chapter will focus on preparing the images for input into classifiers with their numerical and statistical representations.

**Chapter 4**: This chapter covers the first case study of the automatic recognition of abnormal artefacts in the natural images of construction materials of glass façades and concrete surfaces. In the beginning, backgrounds and related works are reviewed, and then the proposed methods for glass façade and concrete surface crack recognitions are introduced. The dataset, the experimental setup, and the results are presented and discussed, and future works are described.

**Chapter 5**: Here, we present the first part of the second case study of border irregularity recognition in thyroid cancer nodules. The methods analysing only ROI points given by doctors are presented in this chapter, while the methods based on analysing the textures around the lesion borderline are presented in chapter 6. We conduct a literature review of the existing works on the border irregularity of thyroid cancer lesions and related works and introduce our datasets. Then the different proposed methods based on borderline distances or FD inspired are introduced. The chapter will also show the effectiveness of the methods based on the ROI points to visualize the irregular regions along the borderline. Then the experimental setup and several evaluation protocols are described before presenting the experimental results, discussions and recommendations at the end of the chapter.

**Chapter 6:** Presents the second border irregularity recognition approach for thyroid cancer using texture analyses of the ribbons around the border. Two methods for building the ribbon around the borderline using the ROI points are presented. The methods utilizing pixel intensities in the ribbon based on ULBP, HOG, and HOL are described in addition to presenting two CNN models. Finally, the experimental results are presented and analysed.

**Chapter 7**: We present decision and score-based fusion schemes of the methods presented in chapters 5 and 6 and build a hierarchical multi-classifier model based on a decision tree (DT). The results of combining two, three, and five methods are presented, compared and analysed.

**Chapter 8:** Concludes the thesis with a summary of the work, discussions, an outline of future works, and research directions.

# Chapter 2: Machine Learning for 2D Image Analysis

This chapter is concerned with providing background information that helps enable the design, development, and testing of the performance of ML and AI algorithms specifically targeting image analysis tasks and applications. In this chapter, we shall first briefly describe the architecture and the main approaches for implementing a typical pipeline of machine learning techniques. We shall then identify the main components of such pipelines but focus on reviewing the main characteristics of members of the two traditional and deep learning classes of such algorithms. For the two case studies of abnormality recognition, we primarily employ traditional ML algorithms to perform the many tasks involved, but we also examine the efficacy of some DL-based approaches. Additionally, we use several of the performance metrics and evaluation techniques mentioned in this chapter throughout the thesis.

## 2.1 Introduction

Image analysis histories can be traced back to the emergence of photography and the realization of its use as a rich source beyond recording events. As camera devices evolved from grayscale to coloured from lower to higher resolutions, computers emerged, and computer-based image analysis became possible through image digitalization. In recent years, image processing has been increasingly used in many application areas, with advances in different image modalities accompanied by advances in computer systems. Nowadays, the application of image analyses covers a vast area of daily life, science, and technology. The applications of digital image analyses are so large that they need to be organized, for instance, by categorizing according to the image sources [5]. Most imaging systems are based on energy sources coming from the electromagnetic spectrum, while others are generated from acoustic, ultrasonic, and electronic energy sources or created synthetically using computers. Some application areas of digital image analysis include; 2D object recognition [6], face identification tasks [7] (e.g. at border control), robot/drone vision [8], and medical image analysis [9] (e.g. automatic disease recognition). Image analysis is increasingly deployed in healthcare, especially with the development of various medical image modalities such as PET (positron emission tomography), MRI (magnetic resonance imaging), and CAT (computer-aided tomography), where multiple X-ray images taken at different angles to build a rich image for assessing bone cracks and ultrasound images (US) for assessment of pregnancy development for instance.

Before the emergence of computer technologies, experts and radiographers manually carried out image analysis tasks. However, advances in computer technologies have led to increased interest in utilizing computer capabilities to support many of these tasks. Initial advances targeted the development of automatic or semi-automatic Computer-Aided Systems to speed up some well-understood and easy-to-model image measurements used in various science, engineering, and medicine areas. Successes in such computerized systems generated new opportunities for mass deployment in more challenging image

analysis tasks and motivated research into using the computer capabilities in processing and automatic detection of certain textural/structural image features that human experts would do manually through a time-consuming error-prone process. The result was mass deployment on the one hand and mounting costs due to the need for expensive machinery and training. In parallel, computer hardware became more advanced with higher computational capabilities and more affordable. However, the burden of mass deployment of image applications highlighted the urgent need for developing smart and efficient software and computational models to exploit the increased computational power of the hardware. This cycle of combined advances in hardware and software generation started to facilitate the development of long-perceived models for AI algorithms, as initially envisaged by Alan Turing and the early pioneers of computer technology. Nowadays, there is more realization of the urgent need to deal with the rapidly growing demand for automatic image analysis by developing efficient and well-designed AI and ML models.

The general purpose of manual or automatic image analysis is to find means of distinguishing/discriminating images or image objects. Such image analysis depends on measuring variation in image data (e.g., intensity values, colour data, edges, shapes, and their statistical parameters), and such measurements are more informative in image regions containing more textural/structural features than smooth regions. Hence, image analysis is akin to texture analysis. Accordingly, AI image analysis algorithms aim to extract or learn application-relevant discriminating texture features.

The main common components of any ML/AI algorithm can be split into two parts: (1) image pre-processing and feature extraction/learning; (2) classifier schemes/architecture, training, and testing experimental work. Image pre-processing is necessary to enable the extraction/learning of the relevant textural/structural features in a consistent manner. Such procedures may include denoising, normalizing illumination variations, segmenting the objects of interest, partitioning the images, resizing the input image to a fixed size, etc. In short, pre-processing is about preparing the input images for the intended automatic analysis consistency.

Two different types of ML and AI algorithms for data and image analysis have emerged over the recent decades: hand-crafted feature-based algorithms and Deep Learning (DL) algorithms. The main significant differences between the various members of these two types are that various specific texture features need to be engineered from the images in the first type. In contrast, the DL algorithms learn different image features automatically that can be used to distinguish different classes of images/objects.

## 2.2 Hand-Crafted ML Models

Traditional handcrafted feature-based machine learning algorithms have been developed and successfully deployed in various image analysis tasks. The general process of any ML handcrafted

feature-based image analysis method starts with image acquisition using any imaging modalities, followed by segmentation of the region of interest (ROI), which can be done either manually or automatically using any segmentation algorithm. Then, texture features using image analysis methods are extracted in the form of numerical values and fed to a classifier as the last step (see Figure 2.1).

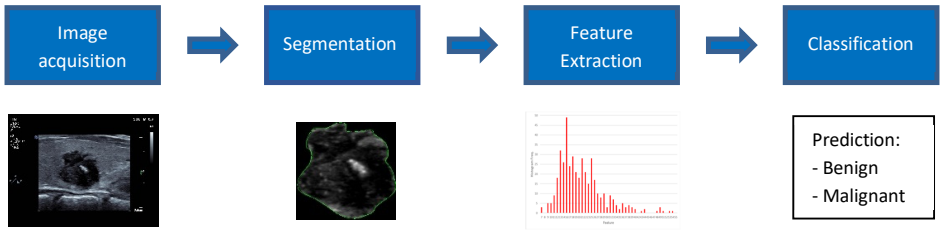



Figure 2.1: Pipeline of typical ML based on handcrafted features.

In the following, we describe some common texture features used in various applications, including cancer analysis from ultrasound images (US), face recognition, Covid-19 detection, etc. We group the works as much as possible based on the same or related feature types.

In [10], several histogram moments, such as mean, kurtosis, entropy, energy, and smoothness, are extracted from histograms of pixel intensities of US images for thyroid cancer benign and malignant classification. In another work, HU's invariant moments, Gabor Wavelet Transformation, and Entropy (Yager's measure and Kapur's entropy) are used to detect benign and malignant ovarian tumours from US images in an automatic CAD system [11]. Hu's invariant moments is a statistical measurement invariant to scale, translations, and rotations used for image analysis. Seven invariant Hu moments are calculated from the normalized central moment of an image and used as discrimination features. Gabor wavelet is another feature extraction method looking at the frequency domain of an image. Gabor works as a bandpass filter, i.e., it allows certain bands of frequencies to pass and reject others. It captures specific frequencies of an image or around a region of an image in a certain direction and can be used with different scales and directions for classification tasks. Texture features such as Haralick and Zernike moments are used in a Covid-19 diagnosis tool, which classifies the input chest digital X-ray image into several respiratory diseases such as COVID-19, viral pneumonia, and bacterial pneumonia [12]. The Haralick [13] texture features like homogeneity and contrasts are calculated from the normalized GLCM matrix and give different aspects of the grey level distribution of the image region under investigation. Zernike moments [14] are orthogonal invariant moments widely used in image shape feature descriptions in many applications such as characters or object recognition and classifications. The most distinctive properties of the Zernike shape descriptor are its invariance to noises and non-redundancy. It is interesting to mention that it is possible to reconstruct an image from a set of Zernike moments. GLCM is a feature characterizing image textures by calculating a histogram of the frequency of a pair of pixel intensities in certain unique geometrical relationships that occur

across the image. Several statistical measures, such as contrast, correlation, homogeneity, and energy, can then be derived from GLCM for recognition tasks.

In another work for thyroid cancer malignancy classification from 3-D high-resolution US images, textural features are combined with the input image's discrete wavelet transformation (DWT) for better performance [15]. Contrast, entropy, and homogeneity texture features are extracted from US grayscale images based on a grey-level co-occurrence matrix (GLCM). Morphological features are extracted from cancer lesion shapes and combined with textural features for breast cancer diagnosis from US images [16]. The morphological features include factor, roundness, aspect ratio, convexity, solidity, and extent and the texture feature is based on autocovariance coefficients from image statistics. In [17], texture features such as standard deviation (σ), Fractal Dimension (FD), and Gray Level Co-occurrence Matrix (GLCM) are used to classify ovarian cancer tumours from 3D US images using a decision tree (DT) classifier. FD usually represents an index of complexity and can be defined as the ratio of the rate of detail change in the pattern with respect to the scale at which the pattern is measured. It finds applications in various fields, including computer vision and image analysis. The FD of an image has shown a direct correlation to the texture roughness in images [18]. Cancer lesion border irregularity can be considered surface roughness in the image since it is manifested in the texture variations around the border and can be analysed using FD for lesion border irregularity recognition.

A histogram of the Local Binary Pattern LBP feature (HLBP) is used to discriminate between benign and malignant ovarian cancer tumours from US images [19] and for pose invariant face recognition [20]. LBP [21] is a powerful texture feature based on local pixel intensity, which is used in our case studies of the construction material cracks and cancer lesion border irregularity recognitions. LBP is invariant to rotation and illumination changes and computationally very fast, and it is a common texture feature in many applications. For each image pixel, a binary LBP code is calculated by comparing the intensities of the actual pixel with its neighbouring pixels. A discriminative feature of Residual Exemplar LBP (ResExLBP) and an iterative feature selection method ReliefF (IRF) are used for Covid-19 detection from X-ray images [22]. ResExLBP is a feature extractor, whereas IRF is a feature selector. The basic idea here is to generate a high-dimensional feature vector using ResExLBP and then select the most discriminative features before feeding them to a classifier. This technique is known as feature relevance or feature importance analysis. Histogram of oriented gradients (HOG) is another texture feature like LBP calculated densely at each pixel and used in many applications such as face recognition [23] and concrete crack detection [24]. In opposite to LBP, it captures texture gradient directionality at each pixel.

The Scale Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF) features are used in face detection [25]. SIFT is a feature that describes the local content of an image and reduces the entire image into a set of locally distinct points (key points) together with the description of the

locally distinct points (descriptor). This type of feature is considered sparse sampling, described earlier since it extracts only key points or pixels from the image. The key points are extracted using the deference-of-gaussian approach, where multiple gaussian image blurring is conducted at different magnitudes. The key points are the extreme points of the subsequent subtraction of the blurred images. On the other hand, SURF is partially inspired by SIFT, creates key points and their descriptors similar to SIFT, and is computationally faster.

In [26], among others, blood vessel tortuosity and contour irregularity of foveal avascular zone FAZ were used for early detection of sickle cell retinopathy (SCR) from 3D images of retinal vascular structures. The tortuosity feature measures the complexity of the texture structures, where a texture is considered more complex the more twists and branches it has. Since the retinal blood vessels of SCR patients are more tortuous than normal vessels, the tortuosity index can be suggested as a discrimination feature. The FAZ's contour irregularity index is calculated from the ratio of the FAZ contour perimeter and a reference circle perimeter. This feature could be of interest for construction material cracks and the cancer lesion border irregularity investigated in this thesis since both show some complex texture structures, including twists and branches.

In our work, we adopted some of these handcrafted features and their performances are tested for our case study applications; however, we also developed new texture features related to our abnormality detections.

## 2.3 Deep Learning Models

Over the last few years, convolutional neural network (CNN), a special type of deep learning, has become a popular tool in many applications, especially in computer visions that deliver the highest accuracies outperforming the traditional machine ML approaches. The applications of CNN in computer vision include object recognition [27], [28], image classification [29] and segmentation [30], natural language processing [31], etc. A typical CNN architecture consists of several convolutional layers and pooling layers sequentially connected for feature extractions and feature dimension reduction and fully connected layers for classifications, as illustrated in Figure 2.2. Deep learning techniques operate in two phases: training and testing, similar to conventional ML algorithms. In the first phase, a model is trained through a back-propagation algorithm [32] using a set of training data where potentially millions of training weights are iteratively updated. In the second phase, the trained model is used for testing unseen data. In each training iteration, different features are extracted through a feed-forward process across different deep learning layers.

The CNN learns different discrimination features automatically and does not need to be engineered, contrary to handcrafted ML algorithms. This advantage comes at the expense of demanding high memory and computational resources and a high volume of training data to deal with the high

dimensionality of the feature space. Recently, Graphics Processing Unit (GPU) development satisfied DL requirements for high computational performances due to its capacity for parallel processing, while high amounts of data have been available through social platforms and other internet activities in certain application domains. CNN models gain further momentum among researchers through exploring different architectures and transfer learning.

Depending on the application domain, different CNN architectures with a different number of layers are used. Generally, the more layers an architecture has, the more complex features can be captured, although having more layers does not necessarily give better performances despite making the architecture more complex and resource-demanding. The training process is lengthy and requires a lot of data and computation resources. On the other hand, testing new data samples is fast since it does not require back-propagation.

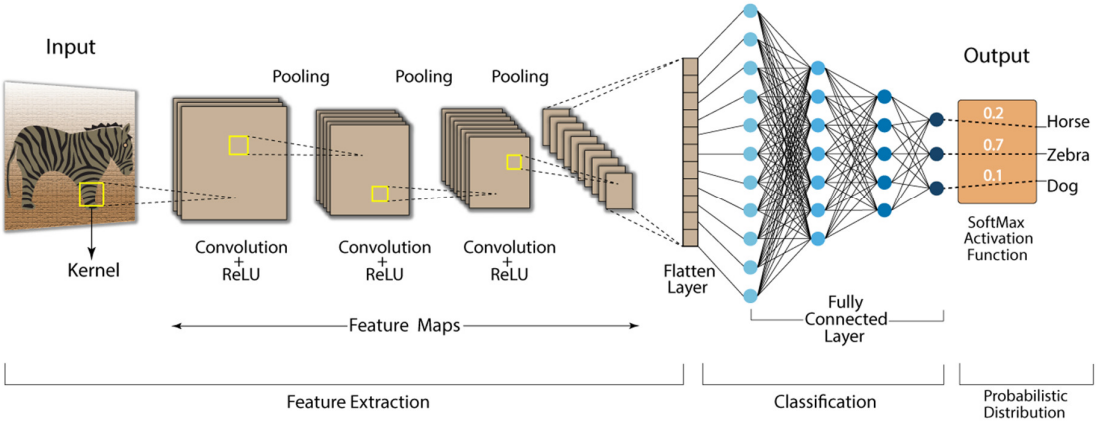



Figure 2.2: Typical CNN architecture [33].

Convolutional layers convolve the input image or the intermediate feature maps using filters of different sizes to produce new feature maps. Pooling layers summarize the feature maps into smaller dimensions to reduce the number of parameters to learn and thus reduce computation needs. On the other hand, the fully connected layers function as a classifier, where each neuron in a layer is fully connected to the previous layer, hence the fully connected name. The 2D feature map from the last convolutional layer is flattened to 1D before it is fed to the fully connected layers. The output activation vector from the last fully connected layer is then turned into predictions through the softmax activation layer (see Figure 2.2).

The problem of demanding a large dataset for training a CNN model is solved to some extent through data augmentations [34]. The basic data augmentation approach is to produce several images from a single original image using straightforward operations such as cropping, flipping, shearing, scaling, rotations, translations, etc. Another widespread way to mitigate the large data requirements for training CNN models is to use transfer learning [35]. The basic idea here is to transfer the knowledge gained

from training CNN on a large available dataset to a new task with a smaller dataset from a different application domain. Contrary to training from scratch, the transfer learning process does not start with random weight initialization, instead starts with already learned weights and fine-tunes them for the new dataset.

One common transfer learning approach is using the pre-trained model as a feature extractor [36]. Here the model has learned enough general features from the large dataset (e.g. ImageNet [37]) and can be used to extract the features from the new dataset without the need to retrain the model. The features can then be used to train classifiers such as SVM, kNN, or new CNN fully connected layers. This approach is faster since it does not require retraining the convolutional layers but might not capture features specific to the new domain. Another common technique is based on fine-tuning the pre-trained model. Here, some or all convolutional layers are retrained starting from the weights already learned through transfer learning. As a result, the model uses the learned features from the already-trained model and new task-specific features learned from retraining the convolutional layers. Generally, the initial convolutional layers capture more generic features, while the later ones capture more specific features. Therefore, it is worth retraining deeper layers while freezing the initial ones. This raises the question of how transferable the features are from different stages of the convolutional layers from the bottom, middle, or top of the CNN layers [38].

Although deep learning, especially CNN, has been showing supremacy over all other traditional ML algorithms, it has the main issue of interpretability, i.e., it is difficult to explain and comprehend their decisions. In an attempt to explain the typical black-box behaviour of deep learning, several concepts for CNN feature visualizations, including Activation Maximization, Network Inversion, and Deconvolutional Neural Networks (DeconvNet), have been conducted recently [39], [40]. The explainability of AI decisions is particularly important when it comes to medical applications. Therefore, using traditional ML algorithms in some medical applications is still desirable, where the features are engineered, and decisions are made in specific explainable steps.

## 2.4 Assessment of ML Algorithms

In the previous sections, we described the two common ML approaches based on traditional and deep learning methods; however, typical approaches for assessing the performance and reliability of such methods are necessary. Therefore, this section is concerned with describing a typical framework of experimental work to assess the success/reliability of ML Algorithms. This framework includes the following component.

- Selecting experimental datasets of images relevant to the investigated task
- Designing experimental protocols regarding the split of training and testing data
- Identifying metrics for the method's performance evaluations

### 2.4.1 Designing Training and Testing Protocols

Given that the aim of building most ML models is to make predictions, it is natural to choose the model giving the best prediction accuracies. In a perfect world, a model that generalizes on unseen data beyond the training data should be chosen. Model generalization is an essential performance measure when ML models are evaluated. The more generalized a model is, the lower its over-fitting. A model is overfitted if it models a particular dataset too well that it fails on a new dataset. Ideally, models are trained with enough data from different natures and sources to minimize over-fitting; however, data is not always available. Therefore, selecting an optimal model that generalizes the best with the least over-fitting is challenging. Therefore, determining how overfitted a model is, implies some compromises when designing a training and testing protocol.

A common practice to show the level of over-fitting is to use separate datasets for training and testing, where the testing dataset is referred to as the external or unseen dataset [41]. There is often only one dataset available; therefore, removing some of the data from the training set for testing is necessary. The selected model, in this case, will be sensitive to the evaluation set if a smaller dataset is used. Another way to split the dataset and create a model less biased to how the dataset is split is to use k-fold cross-validations [42]. This method splits the dataset into equal k sets, where each set is used in one fold for validation and the remaining sets for training. This ensures that each of the k sets is used once for validations. The final model performance is then averaged over all k model performances. Figure 2.3 shows k-fold cross-validation when k=5, whereby 20% of the dataset is usually used for testing and the remaining 80% for training in each fold.

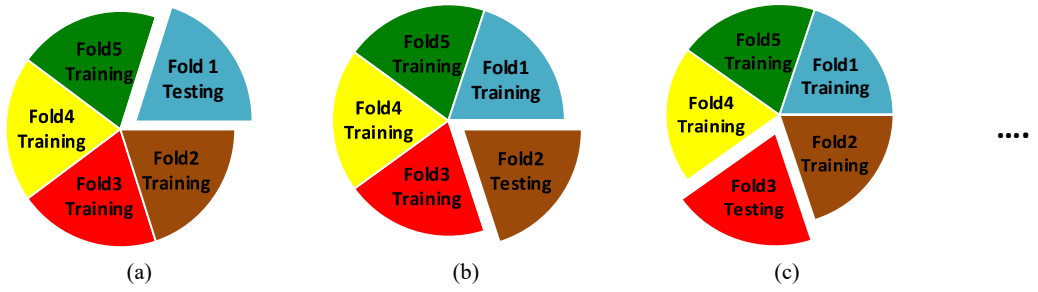



Figure 2.3: 5-fold cross-validation (a) fold1 is used for testing and the rest for training (b) fold2-testing (c) fold3-testing.

A special case of k-fold cross-validation is Leave-One-Out cross-validation (LOOCV) [22], [42], where k is equal to the number of cases in the dataset. Thus, in each k fold, one case from the dataset is left out for validation and the rest for training. Again, the final method performance is calculated by averaging all k-model performances. This protocol provides a much less biased performance measure than other protocols due to fitting the models repeatedly to k-1 instances of the dataset. This comes with the expense of computation time, especially if the evaluation method is complex. However, this method runs into the model selection problem when testing external data since many models are trained.

Another protocol worth mentioning here is the Holdout evaluation method which splits the dataset randomly into training and testing sets. The size of the testing set is usually smaller than the training set and is typically around 20%. The downside of this method is that it may result in a highly misleading performance since the random split is run one time. The last evaluation protocol described here is Monti-Carlo cross-validation (MCCV) [43], which, similar to the Holdout method, splits the dataset randomly into the training and testing sets but repeats the random splitting multiple times. For each split, the model is fitted on a training set and tested on a testing set. The final performance is calculated by averaging over all models. The disadvantage here is that the method may give different performances when repeated with different random splits, in addition to the fact that some of the dataset instances may not be validated while others validated multiple times. This effect can be reduced by increasing the number of random splits. We adopt a similar protocol for method evaluation in our second case study of border irregularity recognition (protocol3 see section 5.5.1.1). Similarly, we split our dataset randomly into training/evaluation sets several times; however, we used a different measure to select one model instead of averaging over all models.

### 2.4.2 Assessing the Performance of ML Methods

One metric for measuring any ML model's quality is its generality. A model is considered generic if it can represent any instances from the class of the training set [44]. Another measure is that a model should not be able to represent an instance of a class the model is not trained on. Another interesting measure of model quality is its compactness or simplicity, which represents the number of parameters used. A model is meant to be compact if it uses as few parameters as possible. The work in [44] is an example of using compactness, specificity, and generality as measures for model quality. The measures are analysed both theoretically and practically to measure the quality of the statistical shape models.

Most of the time, classification accuracy is used as a metric for measuring the performance of any ML algorithm. However, there are many other measures to truly judge the performance of the models, such as sensitivity, specificity, ROC curve, balanced accuracy. How relevant is each of the measures depends on the application task and the nature of the data. In some applications, one may be more interested in higher predictions for one of the classes than the others. In medical applications, for instance, it might be more critical to predict every cancerous lesion correctly but less critical to accurately predict every healthy cancer lesion. In the following, we define some of the widely used measures for AI model performances.

Classification accuracy is defined as the rate of the correct predictions to the total predictions made where the model's overall performance is measured from both classes' prediction rates.

$$\text{Accuracy} = \frac{\text{Number of correct predicted cases}}{\text{Number of all cases}} \qquad (2\text{-}1)$$

Sensitivity or Recall is the measure of the ratio of the number of correctly classified positive cases to all classified positive cases and can be calculated as follows:

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (2\text{-}2)$$

Where TP is the true positive count, i.e., the number of correctly predicted positive cases, FN is the false-negative count, i.e., the number of false predicted positive cases.

Higher sensitivity might be more appropriate than other measures because predicting positive cases is more important than predicting negative cases in many applications, including medical and material fault detection in construction or industry.

Specificity is the rate of correct predicted negative cases to the number of all negative cases and is calculated as in the following:

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (2\text{-}3)$$

Where TN is the true negative rate, i.e., the number of correct predicted negative cases, and FP is the false positive rate which is the number of false predicted negative cases.

A confusion matrix is an n-by-n matrix showing TP, TN, FP, and FN, where n is the number of classes. Figure 2.4 shows a confusion matrix where n=2.

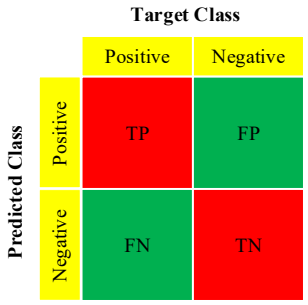



Figure 2.4: Confusion matrix.

The next interesting metric for measuring the model performance is Precision which is the rate of correctly classified positive cases to the number of all positive predictions and calculated as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2\text{-}4)$$

F1-Score is quite important when measuring an unbalanced dataset's performance since it uses both false negatives and false positives, which means that missed cases in both classes are considered. It is calculated as a weighted average of sensitivity (recall) and precision as follows:

$$\mathrm{F1-score} = \frac{2 * (\mathrm{precision} * \mathrm{recall})}{\mathrm{precision} + \mathrm{recall}} \qquad (2\text{-}5)$$

The receiver operating characteristic curve (ROC) is a plot showing the relation between the true-positive rate (TPR) and the false-positive rate (FPR). TPR is sensitivity (or recall), and FPR is (1-specificity). Many ML models are able to predict the probability of a data sample to be predicted as various classes. The data sample is then assigned the class label using a certain probability threshold. The ROC curve (see Figure 2.5) plots TPR against FPR using several probability thresholds. The solid curves indicate the model's performance and the closer the curve to the top left corner, the higher is model's performance. The green line shows the performance of a random classifier model that is not trained and therefore has no discrimination capabilities.

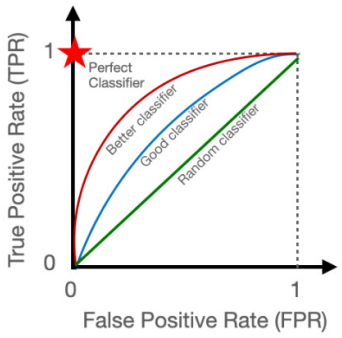



Figure 2.5: ROC curve [45].

Further, the area under the curve (AUC) of the ROC can be used as one value measure of the performance of the model. The higher the value, the better the prediction of the model. In general, models giving probabilities are preferred over models giving only labels because, depending on the application, the probability thresholds can be adjusted, e.g., to meet the application's sensitivity and precision requirements. Finally, Youden's index is another performance metric that emphasises both sensitivity and specificity and is calculated as (sensitivity + specificity -1) [46].

In our two case studies, we used mostly accuracy, sensitivity, and specificity to judge our method performances. In the case of glass and concrete crack recognition, the emphasis was more on sensitivity since it is essential to recognize every broken glass or concrete, but false identifying non-cracked ones might not be as critical. However, in the border irregularity recognition of thyroid cancer, the emphasis was on both regular and irregular border cases by considering the gap between sensitivity and specificity

when evaluating our methods. Thus, Youden's index might be interesting for performance measures in this case study.

## 2.5 The Main Challenges of ML

We discussed various ML assessment methods in the previous section, including training and testing protocols. In this section, the main challenges of machine learning methods are described. Issues with small or unbalanced datasets, ground truth, and data normalizations (different lesion sizes) are investigated.

### 2.5.1 Datasets and Ground Truth

Enough datasets and reliable class labels are crucial for the success of any AI algorithm. Ground truth is the true target class that can be trusted as the gold standard "truth" and can be relied on. Domain experts usually annotate the class labels. Generally, dataset and class labels can be either adapted from an existing dataset or adopted and then modified or newly created. Many free datasets and labels are available from academic research organizations for academic research. Others can be obtained through governmental institutions with some restrictions, while commercial companies offer some licenced datasets for commercial use. However, there are very few publicly available datasets, especially in medical and security application fields, and it is very difficult to collect new datasets due to personal privacy and hospital or governmental restrictions.

In the context of digital image analysis, a dataset is a set of images representing different classes of the problem domain. For instance, in the applications of cancer recognition from medical images, there are two classes of benign and malignant images: the malignant class is positive, and the benign one is negative class. The class labels are usually created by the domain experts, such as experience doctors in the medical field, but they can also be created synthetically by computer algorithms. Also, a mixture of humans and computers can annotate the class labels. In this case, radiologists mark the region of interest by setting several points around the object for the computer algorithms to automatically segment the region of interest. For any image analysis model based on ML to perform well, the image contents must be clearly selected and the labels carefully annotated; otherwise, there is no way to improve the performance. This is particularly important in ultrasound image analysis since, by nature, the image contains (tissues, organs, cancer lesions) has very poor visibility, and therefore it is quite challenging to locate the region of interest. Further, labelling US images is subjective and depends on the radiologist's experience. In this sense, several doctors with different experiences might label the same US image differently. While this is not an issue in our first case study of crack recognition since whether a glass panel is cracked or not can be easily determined visually, it is a serious issue in the second case study due to the inter and intra-observers subjectivity variation of lesion border irregularity.

For robust model building, most ML methods need large datasets for training, especially when it comes to deep learning methods. Therefore, often the size of the original dataset is expanded by augmentation methods, where new instants are created from original images by scaling, rotation, adding noise, flipping, etc. Although the augmented images do not represent the true variation of the images in the application domain, they hopefully increase the robustness of the model. Another issue is unbalanced datasets, where the number of cases in one class is much higher than in the other. This will often lead to high sensitivity of the model toward the majority class. Many ways to balance the training process can be attempted, for instance, by choosing certain training and evaluation protocols (see section 2.4.1) or using different performance measures that consider the accuracies of both classes' sensitivity and precision.

### 2.5.2 Data Preparation

Data preparation and understanding are the first steps in a successful ML project. A dataset for a method operating in a specific domain needs to be identified and analysed. Data preparation involves data cleaning (removing data duplicates and outliers), data transformation (changing the scale of the data suitable for the machine learning method), and Dimensionality reduction (creating compact projections of the data). The data may include duplicates, which do not add more information to the training and need to be removed. Some other data may be wrong and not fall into the application domain. Other data cleaning processes could involve removing artefacts (objects inside the image that are not relevant to the task or object of interest) from images, such as numbers, texts, and crosses, as in the medical US images. Machine learning algorithms work best when the input data, which are numbers, are all in the same range. Therefore data (could be extracted features) needs to be rescaled by processes such as normalization or standardization (see section 3.5). The normalization process rescales all input data into a range between 0 and 1. On the other hand, standardization rescales all input values by estimating the mean and standard deviation of the training data.

Further, input data to a machine learning algorithm could have high dimensionalities. Not all the numbers (features) are discriminative for the intended task and, therefore, can be removed by dimension reduction methods. One of the simplest dimension reduction methods is to look into the variance of the numbers in a certain column in the input data vector. If the data does not change throughout the dataset, meaning it has 0 variances and therefore is not discriminative and can be removed. Other widely used dimension reduction methods, such as Principal Component Analysis (PCA) and Singular Value Decomposition (SVD), are not described here [47].

## 2.6 Summary

This chapter provided a general background of ML approaches and their pros and cons. We introduced two major types of ML methods, handcrafted feature-based and deep learning-based, and briefly

presented some of their applications, especially in the image analysis field. Different evaluation protocols and performance metrics are introduced to assess ML methods. It has been shown that depending on the application domain, some of the evaluation protocols and performance metrics might be preferred over others. Finally, some of the challenges or difficulties facing any ML algorithm are described. Difficulties such as small or unbalanced datasets, ground truth and class label reliabilities, dataset dimensionality, and different scales of the input data are investigated. We will use some of the handcrafted (LBP, HOG, FD) and deep learning (transfer learning) ML methods in our two case studies in this thesis and adopt some of the described ML assessment methods (five-fold cross-validations) and develop new ones to evaluate our proposed methods.

The next chapter focuses on pre-processing techniques, object detections, different texture feature extraction, and their numerical representations, assessing the discrimination power of the features in addition to data normalization.

# Chapter 3: Texture Analysis: Background, Theory, and Procedures

This chapter complements chapter 2 by focusing on the first component of the pipeline framework of AI applications in computer vision, i.e., the preparation of images for input into the classification component, the extraction of the specific texture feature for handcrafted classifiers, and numerical/statistical representation of the extracted texture features. Image features in CNN classification algorithms are learnt through selected architecture; hence, only pre-processing is needed. Accordingly, we discuss the theoretical background and practices of image pre-processing, the type of features to be extracted in the case of handcrafted classifiers, and their representation in suitable computing forms. We shall first discuss different methods of image pre-processing designed to prepare the input image (or regions of interest) ready for extracting texture-related features/parameters. We shall then describe the most commonly found texture features (and their representation) in the literature on computer vision applications for different image modalities. We end with a summary discussion on exploiting this knowledge in the following chapters. We use some of the hand-crafted features, pre-processing, and data normalization methods described in this chapter in our proposed methods for the two case studies of abnormality recognition (see chapters 4 – 6).

## 3.1 Introduction

In the last chapter, we established that in the application of AI image analysis, there is the need for applying a process of initial steps that lead to the output of a computable representation of each input image in the dataset of interest prior to training and testing via the chosen classifier. The nature of this initial process depends on specific classifier-based requirements on the input representation of images of interest. In all cases, it usually consists of a set of pre-processing steps, extracting the texture of interest and representing these features in a format that computers can process. The performance of any computer vision system mainly depends on the extracted feature's quality. An image representation based on a good feature will make the classifier's job easier. However, extracting perfect features is not an easy task. Thus, image analysis methods aim to extract informative and reliable features.

On the other hand, high discriminative features rely heavily on the image's quality. Image quality can refer to perceived image degradations caused by various distortions appearing on the image during image acquisition, transmission, and post-processing. The image quality could be measured subjectively by humans or objectively by automated methods. The objective image quality measures are done by comparing the distorted image with the available original image (full-reference method) or with the part of the available original image (partial-reference method). In most cases, the original image is unavailable; therefore, the quality is measured using only the degraded image (no-reference or blind measures) [48]. The no-reference (blind) method needs to model the statistics of the reference image,

the human visual system, and the distortion type in the absence of the reference image. The statistical features extracted from the input image measure certain distortions such as noise, blocking artefacts, blurring, and fading. Image blurriness, for instance, could affect any texture extraction algorithm, e.g., any method based on reliable edge detection. Medical image quality measurements are challenging due to the absence of reference images, making blind image quality measures particularly interesting. Due to possible variations in the recording condition and the recording imaging devices, there is a need to mitigate the effect of such variations on the extracted features that may render the performance of the chosen classifier unreliable and less robust. The next section describes various common pre-processing operations applied to the images before feature extraction.

## 3.2 Pre-processing Techniques

Almost all image analysis application pre-processing is expected to deal with general issues relating to the effects of different types of noise, blurring, and illumination variance. In applications where the analysis task is related to particular image objects, pre-processing techniques include determining the ROI, object segmentation and detection, and image partitioning. For instance, we need to select the appropriate crack edges and tumour border region construction in our case studies.

Furthermore, image processing techniques are primarily operated in the spatial domain, i.e., the image processing operations are applied directly to the pixels in the x and y coordinates. If features are to be extracted from transformed domains other than the spatial domain, then applying the selected transform is an additional pre-processing requirement. For example, we need to implement Fourier/wavelet transforms to enable working in the frequency domain, but there are also other domains, such as Hough Transform and Radon Transform. Deep Learning models require image resizing and augmentation as the main pre-processing steps. In the following, we introduce the common pre-processing techniques used in digital image processing.

### 3.2.1 De-Noising

Random noises typically consist of sharp transitions in the grayscale intensities overloaded on the original image pixel intensities, significantly deteriorating the image quality. Smoothing (also called averaging) is an apparent operation to counter the effect of sharp transitions that reduce noise. De-noising filters typically include convolving the image with a smoothing kernel, which blurs the image to a certain degree depending on the kernel size. These filters are also called low-pass filters because they reduce high frequencies (sharp transitions in intensities) while retaining low frequencies (slow-intensity transitions) in the image. The source of noises in digital images could arise during image acquisition or transmission. For instance, CCD sensors in digital imaging devices produce an amount of noise depending on the lighting conditions and temperatures. Digital images could corrupt during transmissions due to channel interferences. Different noises like salt & pepper, Gaussian, and speckle

noises can be found in digital images depending on the image modalities and image acquisition devices. Salt & pepper noises occur on the image as randomly distributed black (pixel intensity 0) and white (pixel intensity 255) pixels, usually caused by faulty switching during imaging.

On the other hand, Gaussian noises have a probability density function of normal (Gaussian) distribution, where the pixel values are made up of the original values plus a random gaussian value. Gaussian noises are created from electronic circuits and sensors due to poor illumination and high temperatures. Another type of noise that typically exists in medical ultrasound images is speckle noise (also called granular noise). This type of noise usually arises due to the sensor's environmental conditions during image acquisitions which vastly degrades the quality of ultrasound images. Researchers use various filter techniques to overcome the effect of noises. These filters include, among others, mean, median, and adaptive filters. For instance, periodic noises can effectively be reduced in the frequency domain using notch or Wiener filters.

### 3.2.2 Image Enhancement

Image enhancement is another useful pre-processing technique emphasizing the intensity transitions in the image. Thus, image sharpening operators enhance edges and other intensity discontinuities while de-emphasizing other areas with low-intensity variations, such as spots. Emphasizing the edges and other discontinuations lead to better feature extractions for specific applications. Sharpening filters (high pass filters) are considered the opposite of smoothing filters since they emphasize the sharp intensity transitions (high frequencies) while reducing smooth transitions. Since smoothing can be achieved mathematically through integration, sharpening is an inverse operation accomplished through digital differentiation. Thus, the strength of derivative operator responses at each pixel is proportional to the intensity variation, enhancing the image's fine details while blurring the low-intensity variations. Common image sharpening filters are based on the image's first or second derivatives. Such filters include the Sobel filter [49], which uses first-order derivatives, and the Laplacian filter [50], which uses second-order derivatives. The first filter calculates the rate of the change in the pixel intensity, while the second one uses the rate at which the first derivative of the image changes.

### 3.2.3 Illumination Normalization

Variations in the image's illuminations greatly influence digital image analysis tasks such as object detection and segmentation. The presence of shadows in different regions of natural images as a result of obstruction of the line of view illustrates such variations and impacts segmentation and the quality of extracted texture features. In US images, shadows in images are related to the presence of echoes. Removing or reducing the effect of illumination variations can increase the performance of computer vision algorithms.

A common method of normalizing illumination in natural images is Histogram Equalization (HE), utilizing the pixel intensity distributions in an image. An image's intensity histogram represents the probability of intensity values occurring in an image. Histogram equalization evenly redistributes these intensity probabilities in the image, utilizing the full range of intensities to increase the image contrast. It can achieve high contrast enhancements, especially in degraded images with a narrow histogram of intensities. This is accomplished by spreading out the narrow density of intensities of the lower contrast areas and thus increasing their contrast. The effect of HE can often be measured by comparing it with a reference image of good uniformly distributed illumination. Applying histogram equalization must be linked to the quantitative measure of illumination variation; hence, some images may not need this pre-processing measure [51].

Interestingly, histogram equalization is used for contrast enhancement in US images using Local histogram equalization (LHE) [52]. LHE is applied on certain sub-blocks of US images depending on the sub-blocks amount of information measured using Entropy. The LHE is applied to the block's centre pixel, given that the block entropy is higher than a certain threshold.

### 3.2.4 Fourier Transformation

In many applications, including digital image processing, it is easier and faster to perform certain tasks, such as filtering in the frequency domain (e.g., Laplacian filter) rather than in the spatial domain. Fourier transformation is one of the common methods for transforming different signals and 2D images into the frequency domain. It decomposes a complicated signal into a sum of a series of simple sine and cosine signals with different frequencies and amplitudes. The approximation of the original signal is better the more signal components with higher frequencies are added. Thus, Fourier transformation allows for studying complicated signals in their frequency content (frequency domain). Since computer vision processes digital images, Discrete Fourier Transformation (DFT) is used to transform images into the frequency domain. DFT needs vast computational power due to many multiplications and additions. Fast Fourier Transformation (FFT), discovered in 1965, is the method to drastically reduce computational needs, making work in the frequency domain practical for many applications, including computer visions. FFT transformed signal or image is also referred to as the frequency spectrum of the image, which includes complex numbers for each frequency component. Absolute values of the complex numbers represent the amplitude of the individual frequency components, which shows the contribution of the frequency component to the original signal. An essential property of Fourier transformation is its inverse transformation, i.e., the signal can be converted to its frequency domain and then converted back to the original spatial domain without any losses. This property allows, for example, to transform an image into the frequency domain to apply a low pass filter to remove fine details (high frequency) and then reverse back to the spatial domain, which is basically represent an image smoothing (blurring) operation.

Following is the Discrete Fourier Transform (DFT) equation for a one-dimensional discrete signal [5]:

$$F_m = \sum_{n=0}^{M-1} f_n \, e^{-i2\pi mn/N} \qquad m = 0,1,2 \ldots, M-1 \tag{3-1}$$

Inverse Discrete Fourier Transform (IDFT) for a one-dimensional discrete signal [5]:

$$f_n = \frac{1}{M} \sum_{m=0}^{M-1} F_m \, e^{i2\pi mn/M} \qquad n = 0,1,2 \ldots, M-1 \tag{3-2}$$

In our work, FFT is used in the second case study to capture the irregularity of the border of the thyroid cancer nodule. However, FFT is applied on a one-dimensional signal formed from borderline distances instead of the 2D image or ROI (see chapter 5).

### 3.2.5 Region of Interest Segmentation

The sub-region in an image where the object of interest lies is called the region of interest (ROI). The quality of any features extracted from the ROI depends directly on the segmentation accuracy; therefore, accurate segmentation of the ROI is crucial for any computer vision applications. Hence, image segmentation is one of the major domains in computer vision. Instead of handling the entire image, it will be more efficient for the image analysis algorithm to process only the region of interest. Image segmentation approaches are based on two properties of discontinuity and similarity of pixel intensities. In the first approach, the image is segmented based on abrupt pixel intensity changes, while the second approach partition the image into several regions based on predefined similarity criteria [53]. Examples of methods for the first approach are edge detection, while for the second approach are thresholding, region growing, region splitting, and merging. Better segmentation performances can be achieved by, for instance, combining the two approaches of edge detection and thresholding. Other segmentation methods are based on, for example, clustering, superpixels, and morphology. Most segmentation techniques pre-process the US image before the segmentation, including removing unwanted artefacts and noises and enhancing contrast for more accurate segmentations [54].

These pre-processing techniques are an interesting extension of our work on border irregularity recognition from US images to improve the quality of extracted features and increase the method's performance. We use edge detection in our first case study to extract crack segments, while in our second case study of tumour border irregularity recognition, we use a simple border approximation from the ROI points instead of cancer nodule boundary segmentation.

## 3.3 Object Detection

Object detection is an essential and challenging task in the field of computer vision, dealing with identifying and locating various objects in an image. Unlike image recognition/classification, object

detection identifies an object's class and location in an image. While image recognition assigns a label to the entire image, object detection draws a bounding box around the detected object and assigns a label. Thus, object detection provides more information about the image and helps understand and analyse scenes in the image or video. Object detection is divided into two main approaches: traditional ML and deep learning-based [55]. In traditional ML-based methods, various features such as object edges, corners, and colours are extracted from the image to identify groups of pixels belonging to a particular object. Then, a regression model is used to locate an object based on the extracted features. Most ML-based object detection methods use local features with a descriptor, such as SIFT and HOG or use local features without descriptors, such as Harris corners [55].

In contrast, deep learning-based object detection methods extract features automatically using an encoder. The encoder output is passed to a decoder to locate an object and assign a label. Deep learning-based methods have become state-of-the-art approaches for object detection, including R-CNN [56], YOLO [57], and SqueezeDet [58].

### 3.3.1 Edge Detection

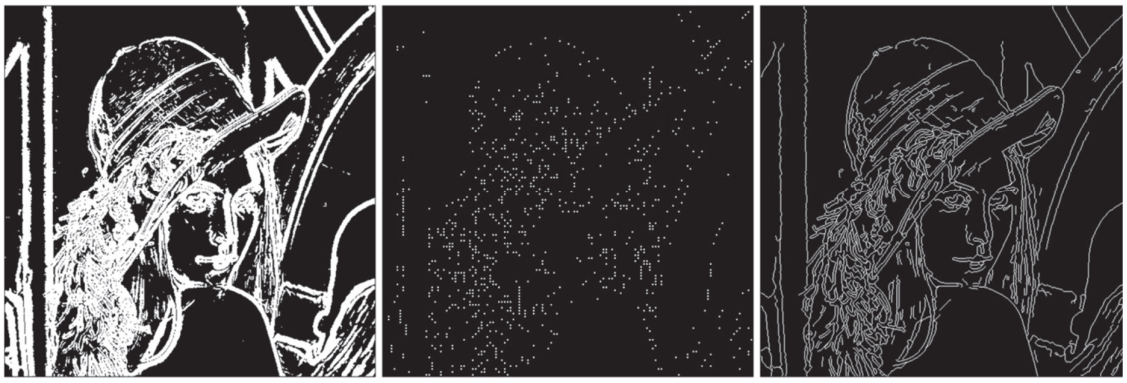
Object boundaries inside an image include valuable information about the object shape used in various image analyses. Image edge detection algorithms can determine object edges by assessing abrupt changes in the pixel intensities around the edges. Edge detection can be considered an alternative for image segmentation and has the additional advantage of drastically reducing the amount of redundancy in image data since only the edge pixels or coordinates are used for subsequent image analyses. Template matching and differential gradients are the two common approaches for edge detection. The two approaches generally look for local maxima of the intensity gradient to locate the regions with high-intensity variations, which is usually the case at the edges. The next step will be to threshold the gradient map of the image or search for the local maxima of the gradient map to get the edge segments. Both approaches use different convolving masks to locate the local intensity gradients. The known template-matching edge detectors are Kirsch and Robinson [59], [60], while differential gradients are Sobel [49], Roberts [61], and Prewitt [62]. Some methods use two convolutional masks in the x and y directions; others use up to 12 convolutional masks, while Laplacian edge detectors use one convolving mask (kernel) [50]. The Canny edge detector [63] is a commonly used, highly effective, and complex edge detection algorithm compared to other methods.

The state-of-the-art ED [64] algorithm uses a smoothing pre-processing step with subsequent gradient map building, like most edge detectors. In contrast to other edge detectors, it picks anchor points from the image and connects them by a smart heuristic edge tracing process. Since the ED edge detector is used as a pre-processing step in several of our proposed methods (see chapters 4 and 6), we describe the algorithm in detail in the following.

First, a gradient magnitude map G is determined by calculating the magnitude of the gradient at each pixel P(x,y) in terms of partial derivatives in the x (Gx) and y (Gy) directions:

$$G = \sqrt{Gx^2 + Gy^2} \quad (3\text{-}3)$$

Then, a gradient direction map is built by comparing each pixel's horizontal and vertical gradients. If the horizontal gradient is bigger or equal to the vertical gradient, then a vertical edge is assumed to go through the pixel; otherwise, a horizontal edge is considered. The weak gradient points are eliminated by thresholding the gradient map. The next step is to select anchor points at the local gradient extremums, which can be visually shown as the mountain peaks on the 3D gradient map. These peaks can be simply located at the edge crossings from the gradient direction map. In deciding whether a pixel is an anchor point, the gradient of the pixel is compared with its two vertical or horizontal neighbours depending on if a horizontal or vertical edge passes through the pixel by examining the gradient direction map. Thus, if a horizontal edge goes through the pixel, the comparison is made with its two vertical neighbours and vice versa. Finally, if the gradient difference between the pixel and both of its neighbours is bigger than a certain threshold, the pixel is considered an anchor point. The higher the threshold, the lower the number of anchor points; consequently, the less detailed the anchor map is and vice versa. The step size for the anchor point scans can vary, but it is kept below radius 5 of the Gaussian smoothing kernel to avoid ambiguous edge maps.



(a) (b) (c)

Figure 3.1: (a) Thresholded gradients cluster, (b) computed anchor points, (c) final edge map produced by ED using anchor points [64].

The anchor map shown in Figure 3.1 is obtained using an anchor threshold of 8 and an anchor interval of 4 (every 4$^{th}$ row and column). The next step is to connect the anchor points, which can start at any anchor point. Figure 3.2 below illustrates the process. The connecting process proceeds to the left and right if a horizontal edge passes through the anchor point and up and down in the case of a vertical edge. Only three direct neighbouring pixels are considered during the move where the pixel with the highest

gradient value is connected (see Figure 3.2). The condition for stopping the connection process is that either the actual pixel is already identified as an edge point, its gradient value is 0, or there are no more pixels. Finally, some of the short edge segments, including a specific number of anchors, can be dropped to enhance the resulting edge map.

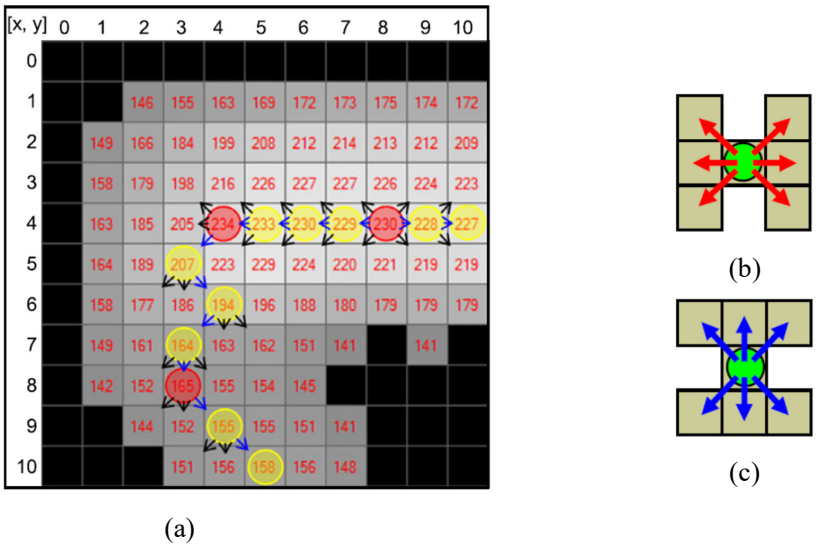



Figure 3.2: (a) An illustration of the smart routing procedure, (b) Horizontal proceeding, (c) Vertical proceeding [64].

Contrary to edge detectors like Canny or Sobel, the ED algorithm outputs the adjoint edge segments (blob of edges) as separate linear pixel chains, i.e., the edge segments are extracted into separate vectors making subsequent feature extraction (e.g., linearity, curvature, connected pixel configurations) easier. ED parameter fine-tuning is necessary for cases of low resolution where the edges are smooth or include high noises. For these cases, a lower gradient magnitude threshold must be chosen to select the anchor points.

In our first case study of abnormality recognition, for instance, the breaks or crack fragments on the glasses and concrete surfaces can be detected by applying the edge detector. In our second case study of thyroid cancer irregularity recognition, accurately detecting the lesion boundary using edge detection and consecutive assessment of the edges may help to recognize border irregularity. However, the poor intensity variation in US images has to be considered in such cases since edge detectors heavily rely on variations in the pixel intensities.

## 3.4 Texture Feature Extraction and Representation

A typical step after image segmentation is feature extraction to allow further image analysis. Features can be extracted from edges, shapes, or object boundaries or directly from an image's colour/grayscale intensities and used for various image analysis tasks such as object detection and recognition. Feature detection and feature description are typical steps in feature extractions. Features can be detected using

entire images, regions, or object boundaries, whereas feature description assigns a numerical attribute to a detected feature. A corner in an image might be used as a feature and described by its orientation and location. Textures are special features appearing on the surface of physical objects as intensity variations with a specific repeated pattern caused by physical surface properties such as roughness and light reflections that allows recognizing or segmenting regions having certain texture characteristics. Thus, textures provide rich information about an image's object or region of interest. The intensities or textures can vary rapidly, slowly, with high or low regularity, and in different directions making texture representation one of the fundamental and challenging problems in computer vision and pattern recognition.

Humans recognize texture in an image when they notice a sudden discontinuity in the pixel intensity/colour values as they scan through the image in different directions. Connected components of pixels with such discontinuities could be linked to an object's edges, corners, or boundaries. After all, images with no discontinuity in their pixel intensity/colour values do not convey any meaningful information. In this respect, we need to remember that the human vision system deals with natural images captured by cameras that sense lights in the visual sub-band of the electromagnetic spectrum with rather limited frequency ranges. However, computer vision systems can analyse images sensed by other special sensors sensitive to other electromagnetic spectrum bands, including thermal cameras sensing heat emission. Machines are also capable of processing/analysing images captured by ultrasound devices that use sound waves to visualize internal human/animal tissues. The same concepts of textures apply to such images and help understand, interpret, and analyse their contents.

Texture-based image analysis is used in a wide range of applications, including medical image analysis, image quality measurements, face recognition and biometrics etc. Poor features will lead even the best classifiers to fail to perform well; therefore, extracting powerful texture features is crucial for successful classifications. The survey in [65] shows an evolution of the texture classifications since 1962 and covers methods from Bag of Words to CNN. Texture analyses are divided into two common texture descriptor approaches for image analysis: sparse texture descriptors involving only specific key points from the image and dense texture descriptors analysing each pixel in the image. In the first approach, reliable detection of sparse regions or points of interest in an image is conducted using methods such as the Harris affine detector [66] and the Laplacian blob detector [67]. Sparse sampling can produce a compact feature vector much smaller than the total number of pixels in an image. However, it can be less useful for many texture classification tasks due to the loss of essential texture content caused by sparse sampling [65]. Therefore, dense texture descriptors are more common methods for texture recognition. Gabor wavelets are one of the widespread dense texture descriptors looking for certain frequency-contain in different directions around the point of interest [68]. The next interesting dense texture descriptor is Local Binary Pattern (LBP), which obtains features by intensity comparison of pixel neighbourhood [21].

Some other popular texture descriptors include LM Filters [69], S Filters [70], Basic Image Features (BIF) [71] (similar to LBP), and Histograms of Oriented Gradients HOG [72]. Some of these texture descriptors can be combined for better performance; for instance, Gabor filters can be served as pre-processing of an image, followed by LBP texture analysis to produce more robust texture features [73]. Further, for highly periodic textures, as in the case of textiles, it is natural to analyse the texture in the frequency domain through Fourier transformation [74]. Fractal dimension-based texture features are alternative to the two dense and sparse texture descriptors, dealing with measuring textures using different scales first proposed by Mandelbrot [75]. In a later development, a MultiFractal Spectrum (MFS) method for texture description invariant to the viewpoint, nonrigid deformations, and local affine illuminations is proposed [76].

In this thesis, we use some of the texture features based on dense sampling approaches such as LBP and HOG and extract other sparse sampling features such as linearity and connected pixel configurations from edge segments.

### 3.4.1 Overview of Texture Feature Extraction Methods

Texture analysis approaches can be categorized into structural, model, statistical, and transform-based methods. **Structural** (or geometrical) based texture analysis assumes texture is composed of many texture primitives (elements) arranged according to particular placement rules [77]. The method utilizes the geometric properties of the texture elements by extracting the placement rule to describe the textures. Texture primitive properties such as average element intensity, area, perimeter, eccentricity, orientation, elongation, magnitude, compactness, Euler number, and moments are successfully used in many applications [78], [79]. A histogram of texture primitives is used for texture analysis to get scale and rotation-invariant texture features [80]. In another work, structural texture features, for instance, are extracted from line structures using Hough transform [81]. **Model**-based methods are represented as a probability model or a linear combined set of basic functions. Model coefficients are used to characterize textures; however, their estimations often require complex computations. Markov random fields are one of the model-based texture analysis methods used in various image processing applications such as texture classification [82], [83]. Fractal-based models are another texture analysis method based on the fractal concept introduced by Mandelbrot [75] (see section 3.4.4). The method is instrumental in modelling many natural surfaces with their roughness and self-similarity properties.

**Statistics**-based texture analysis is one of the early methods proposed in computer vision to facilitate the spatial distribution of pixel intensities by describing the texture features using a set of statistics [77]. Many existing methods based on statistics are based on Julesz's findings on the human visual perception of texture [84]. The human vision system uses feature statistics to recognize textures, including first, second, and higher-order statistics. Some of these statistics-based texture analyses include, among others, moment invariants [85], feature distribution methods [86], and high-order statistics [87].

Hu is the first researcher to introduce moments for 2D image pattern recognition [85]. Moment invariants remain unchanged under translation, scale, and rotation changes. Equation (3-4) below calculates the moment of order (p+q) for a two-dimensional image [88].

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q f(x,y) \quad (3\text{-}4)$$

Where $f(x,y)$ is the image function, and M and N are the image dimensions, $x, y$ are the pixel coordinates, and $m_{pq}$ is a moment of order (p+q).

Central moments of order equal to (p+ q) are then derived from equation (3-4) as follows.

$$\mu_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x-\overline{x})^p (y-\overline{y})^q f(x,y) \quad (3\text{-}5)$$

where $\bar{x}$ and $\bar{y}$ are coordinates of the centre of mass.

Some common first-order statistics, which can be extracted from histograms of different texture features, are presented in the following.

Mean (μ) (first moment) is the average pixel intensity over the whole image:

$$\mu = \sum_{i=0}^{G-1} ih\,(i) \quad (3\text{-}6)$$

Where $h\,(i)$ ith pixel intensity in the image, i is pixel index, and $G$ is the total number of pixels in the image.

Standard deviation (σ) (second moment) measures the amount at which pixel intensities differ from the mean:

$$\sigma = \sqrt{\sum_{i=0}^{G-1} (i-\mu)^2 h(i)} \quad (3\text{-}7)$$

Entropy (H) is the measure of randomness (or disorder) of the pixel intensities in the image and can describe disordered textures. Thus, it measures histogram uniformity:

$$H = -\sum_{i=0}^{G-1} h(i) \log_2[h(i)] \quad (3\text{-}8)$$

Skewness ($\mu_3$) (third moment) measures the shape of the histogram. The skewness value 0 represents a symmetric histogram around the mean:

$$\mu_3 = \sigma^{-3} \sum_{i=0}^{G-1} (i-\mu)^3 h(i) \quad (3\text{-}9)$$

Kurtosis ($\mu_4$) (fourth moment) measures the flatness of the histogram shape:

$$\mu_4 = \sigma^{-4} \sum_{i=0}^{G-1} (i-\mu)^4 h(i) - 3 \quad (3\text{-}10)$$

It is worth mentioning that these statistics can be extracted from histograms of any feature beyond pixel intensities. For instance, in our crack recognition case study, the moments are extracted from the histogram of linearities extracted from edge segments (see section 4.4.2). The above moments are statistics that are simple to calculate; however, they are limited in their texture characterization [89]. Therefore second-order statistics such as co-occurrence matrix using joint probability of pair pixel intensities is also common for texture analysis [90], which is not attempted in this thesis.

The final texture analysis method is a **transform**-based method operating in a domain different from pixel intensities' spatial domain, such as the frequency domain. To transfer an image into the frequency-domain different transformers such as Fourier [91], Gabor [92], and wavelet [93] are used. Hough transform is another method to transform an image into a polar coordinate (Hough space) to detect objects such as lines and circles in an image [94]. The Fourier transformation is used in our work to develop a new method for border irregularity recognition described in section 5.4.10. Hough transform can be interesting for detecting line-like crack segments in our first case study of abnormality recognition.

In the following sections, we describe the features based on LBP, HOG, and FD in more detail since they are used in our two case studies. We need to remind the reader that in addition to these common features, we introduced other new texture features associated with abnormal artefacts under investigation that are sparsely sampled only from edge segments.

### 3.4.2 Local Binary Pattern

Local Binary Pattern (LBP) is a local texture descriptor considered dense sampling since it is extracted at each pixel by comparing its grayscale intensity with neighbourhood pixels proposed by Ojala [21]. The feature is used in both our case studies of cracks and irregularity recognition; therefore, it is described in detail. A rotation invariant version of LBP was introduced by Pietikäinen [86], followed by Uniform LBP (ULBP) by Ojala [95], and successfully used for more than two decades in many application areas [96], such as defect recognition of steel surfaces [97], buildings glass and concrete inspection [36], [98], face recognition [99], and medical image analysis [100].

The method generates a binary code for each pixel in the input image by comparing the intensity of the actual pixel (central pixel) with its eight neighbours (see Figure 3.3). The binary value one is set if the grey value of the neighbour pixel is greater or equal to the grey value of the central pixel; otherwise, a binary value of 0 is set. Repeating this comparison for all eight neighbouring pixels generates an 8 bits binary code representing the LBP code for the current pixel (see Figure 3.3). Generating this code for each pixel forms the LBP representation of the entire image. Since the length of the binary code is 8 bits, the number of possible LBP codes is 256, which implies building a 256-bin LBP histogram as a feature vector for classification.

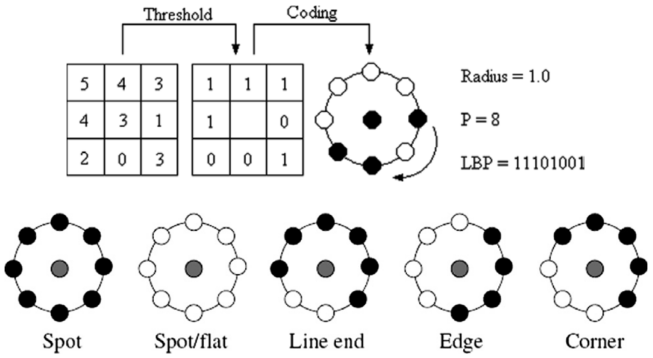



Figure 3.3: LBP code generation for different texture representations [101].

The generalized version of LBP puts no limit on the size of the pixel neighbourhood, i.e., there is no limit on the distance between the centre and neighbour pixels (Radius or R) and the number of neighbours (P) (see Figure 3.4). Extending the distance of the central pixel to its neighbours might capture large-scale texture features in some images [102]. A larger spatial area captured by LBP can be achieved by changing the R and P parameters, as in the following Equation [95].

$$LBP_{P,R} = \sum_{p=0}^{P-1} S(g_P - g_c)2^p \qquad (3\text{-}11)$$

Where $R$ is the radius, $P$ is the number of pixel neighbourhood, $g_c$ and $g_P$ are pixel intensities of the central and neighboring pixel, respectively.

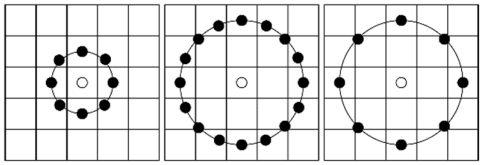


Figure 3.4: LBP using different (P, R) values (8, 1), (16, 2), and (8,2).

An LBP-based method needs to (1) describe the different local patterns and extract them, (2) select a subset of these codes to represent textures (3) use the selected pattern as an effective texture descriptor.

One issue with LBP is how to choose a subset of LBP codes since different LBP local patterns capture different texture patterns. A specific type of LBP called Uniform LBP (ULBP) makes up around 90% of all the LBP codes in natural images [95]. The unique property of ULBP is two transitions from the binary value 1 to 0 and from 0 to 1, producing a total of 58 codes. Using the histogram of the ULBP instead of the LBP may or may not affect the discriminative performance but reduces the feature vector dimensionality drastically and consequently reduces the curse of dimensionality [103]. It is known that 0 and 255 codes capture dark and light spots, while ULBP code groups of 5 ones capture corners, four ones capture edges, and six ones capture line ends (see Figure 3.3). Therefore, it is natural to test individual ULBP groups for their discriminability for different applications and certainly can be combined for better discrimination. Our two case studies will investigate the discriminability of the individual ULBP code groups.

A histogram counting the frequency of each 256 LBP code throughout the image or region is used to discriminate different image textures. However, using ULBP with its 58 codes will perform equally well as reported in many research works [86], [95], [101] and yet computationally much faster due to the reduction in the feature vector dimensionality. The 58 ULBP codes can be combined with the non-uniform codes, for instance, by using all remaining non-uniform codes as one bin added to 58 ULBP histograms as the $59^{th}$ bin [104]. Further, the ULBP codes 0 and 255, representing dark and light spots, can also be removed to form a 56-bin histogram, which may lead to better performances depending on the application. After the original version of LBP, a generalized multi-resolution variant of LBP was introduced by Ojala [95]. Since then, many variants of LBP have been proposed by researchers [105]–[108]. Other works inspired by LBP, such as higher-order local derivative patterns (LDPs) [109] and local directional derivative patterns (LDDPs) [110], have also been proposed.

LBP is invariant of illumination and rotation changes and computationally very fast, making it in particular important property for our first abnormality recognition case study of crack recognition since the glass and concrete datasets are collected in different illumination settings (weather and daytime) and taken by a camera or the drone in different angles.

### 3.4.3 Histograms of Oriented Gradient

The Histograms of Oriented Gradient (HOG) is a dense local feature descriptor originally designed for pedestrian detection [72] and used later for various object detection tasks, including vehicle detection [111] and concrete surface crack detection [24]. Unlike the LBP, HOG captures the gradient and its directionality at the pixel level making it one of the feature descriptors. The basic idea is to divide the input image into equal spatial regions (cells), with the successive building of histogram of intensity gradient and edge directions over the cell pixels and then to concatenate the histograms over the whole image. For better illumination invariance, a local histogram (energy) is calculated from a larger region (called blocks) (see Figure 3.5) and used for contrast normalization of all cells in the block. The block's

combined feature vector is then used as input to a classifier. HOG uses partial derivatives in both horizontal and vertical directions to inspect the gradient of the current pixel instead of comparing pixel intensities as in the LBP method.

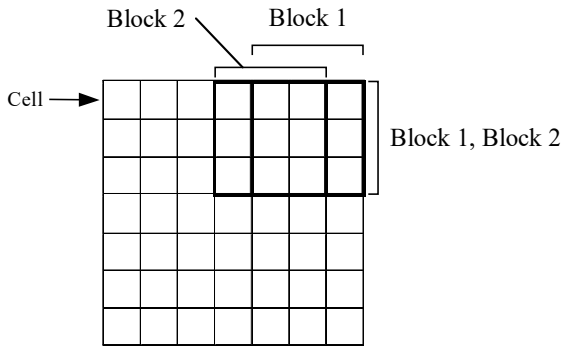



Figure 3.5: HOG cells and blocks.

As mentioned in the original paper, the image is resized into 64x128 pixels to make the division of the image into 8x8 cells and 16x16 blocks easier [72]. The next step is to calculate the gradients at each pixel of the input image by using partial derivatives (g$x$, g$y$) in horizontal and vertical directions as it is given by the equations below.

$$\mathrm{g}_x(\mathrm{i},\mathrm{j}) = \frac{\partial f(i,j)}{\partial x} = f(i+1,j) - f(i-1,j) \quad (3\text{-}12)$$

and

$$\mathrm{g}_y(\mathrm{i},\mathrm{j}) = \frac{\partial f(i,j)}{\partial y} = f(i,j+1) - f(i,j-1) \quad (3\text{-}13)$$

Where $\mathrm{g}_x(\mathrm{i},\mathrm{j})$ and $\mathrm{g}_y(\mathrm{i},\mathrm{j})$ are derivatives at the current pixel with coordinates $(\mathrm{i},\mathrm{j})$.

Next, the gradient map presented by magnitude and orientation (g, $\theta$) is calculated from the gradient values in polar representation using the following equations:

$$\mathrm{g}(i,j) = \sqrt{\mathrm{g}_x^2(i,j) + \mathrm{g}_y^2(i,j)} \quad (3\text{-}14)$$

and

$$\theta(i,j) = \arctan\left(\frac{\mathrm{g}_y(i,j)}{\mathrm{g}_x(i,j)}\right) \quad (3\text{-}15)$$

A histogram counting the magnitudes of different gradient orientations is formed in the next step. Nine bin histograms over orientation (180°/9 = 20°) on histograms x-axis and the sum of the magnitudes on

the y-axis is formed for each cell. Then, the cell histograms are concatenated to build block histograms; for instance, if we build a block containing 4 (8x8) cells, a block histogram of 36 (4X9) bins can be formed. Normalizing the block histogram is done by dividing each bin by the square root of the sum of squares of all bins. The normalized block histogram here is referred to as the HOG descriptor [72]. Finally, all block histograms are concatenated to build a feature vector for input into a classifier.

HOG is used as a texture feature extraction method for our case studies on crack recognition in glass façades and concrete surfaces and thyroid border irregularity recognition. However, they are used slightly different than in the original HOG version (see sections 4.4.5 and 6.3.2)

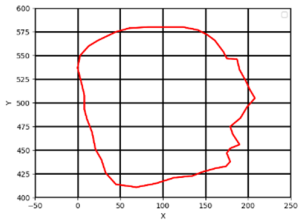
### 3.4.4 Fractal Dimensions

Mathematician Benoit Mandelbrot firstly introduced Fractal dimensions (FD) based on his 1967 paper on self-similarity [75]. The property of self-similarity is used in Fractal geometry to describe complex objects found in nature, explored by the same researcher [112], [113]. Since then, the idea has been used in many applications, including skin lesion irregularity measurement [114] and FD analyses of thyroid cancer lesions from US images [115]. Different methods exist for calculating FD, but they all follow the basic steps of (1) measuring the object boundary or shape using different scales, (2) drawing a regression line of the measured quantities versus the scales, and (3) deriving the FD from the slope of the regression line [116]. One of the common methods of calculating FD is the box-counting method described in the following section.

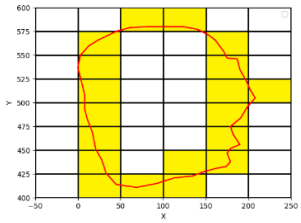
**FD based on Box Counting Method:**

The box-counting method first introduced by mathematician Minkowski (Minkowski–Bouligand dimension, also known as Minkowski dimension or box-counting dimension) is one of the most commonly used methods for calculating FD and is used in various applications [115], [117]–[120]. In this method, the shape (or its boundary) is covered with a grid of equal boxes of certain sizes. Then, the boxes overlapping with shape are counted. The measurements are repeated using boxes of different sizes or scales (the length of one side of the box). The FD is derived from the relationship between the number of boxes overlapping with the shape and box dimensions (scales) using equation 3-15 below. Assume N($\varepsilon$) is the number of boxes using a box side length of $\varepsilon$, then the FD is defined as the slope of the log-log relationship between N($\varepsilon$) and the scale $\varepsilon$.

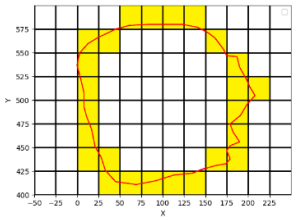
$$D = \lim_{\varepsilon \to 0} \frac{\log N(\varepsilon)}{\log \varepsilon} \qquad (3\text{-}16)$$



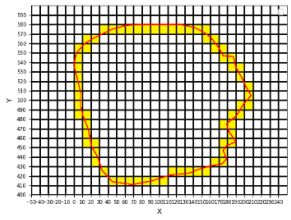
(a) X and Y axis are the coordinates of the boundary points.



(b) Yellow-coloured boxes are overlapped with object boundaries.



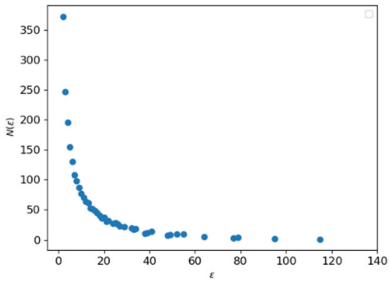
(c) Using box sizes twice as small as the previous one.



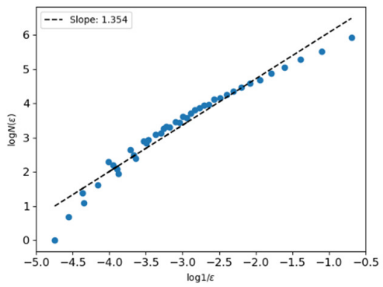
(d) Using box sizes twice as small as the previous one.

Figure 3.6: FD using the box-counting method.

Figure 3.6 above shows a grid of boxes of three different sizes, where the overlapped boxes with object boundaries (red curve) are marked in yellow. In practice, the box scale starts with half of the image width or height, i.e., the number of boxes is 4. Then the size is divided by 2 for each scale until the scale of 2 pixels is reached. Figure 3.7 a below shows the number of boxes recorded for each of the box scales. The number of boxes increases as the box scale decreases. The changes in the number of counted boxes are slow for the bigger scale, and it is getting faster when approaching smaller scales. Then, the regression line (dashed line Figure b) is used to fit a line to the log N($\varepsilon$) and log 1/$\varepsilon$, where the slope of the regression line represents the FD value.



(a) Relation between scale and the number of boxes.




(b) The slope (or FD) of the regression line is 1.354.

Figure 3.7: Fractal dimensions using counting boxes. The plot shows the relation of the number of boxes to the box scale.

The FD using box counting described above is investigated for border irregularity assessment of cancer nodules, and a new method inspired by FD is introduced.

### 3.4.5 Assessment of Texture Feature Discrimination Power

This section describes some approaches to test the application-related discriminating capability of selected texture features. In any computer vision recognition task, the powerful features are those having greater distances in the feature space for different classes. Thus, good discriminative features have similar values for samples of the same class and different values for samples of different classes. In most cases, it is hard to check the discrimination power of the extracted features before using them in the classification experiment. Reducing the number of redundant features reduces the computational cost and improves the model's performance in many cases. Thus, irrelevant features can mislead the classifiers resulting in lower performances of the classification algorithm. Therefore, methods for feature selection are essential for determining the most discriminative features for the target class from the input feature set. Further, feature selection can reduce the overfitting of prediction models trained on small datasets [121].

Two common methods for feature selection are the wrapper method which uses the classifier for the feature selection, and the filter method, which is independent of the classifier [122]. While The wrapper method involves an optimized predictor or classifier in the feature selection process, the filter method uses general characteristics of the training set to select the most relevant features. The general characteristics include correlations between input features and target class, whereby a correlation-dependent score is assigned to the input features to rank them and to select a subset with the highest scores as the input feature vector. Recursive Feature Elimination (RFE) is one of the wrapper methods that seek a combination of predictors that maximize the model performance by the process of adding and removing predictors. RFE was initially used for gene selection for cancer recognition utilizing SVM and later used as a popular method for feature selection [123]. Some classifier algorithms use feature selections as a part of their learning process, including Lasso [124] as one of the penalized regression models and random forest [125], which ensembles multiple decision trees and uses Gini and permutation indexes to determine feature relevance and so reducing the number of features. Finally, dimensionality reduction methods are another way to reduce the number of features. Unlike feature selection, dimensionality reduction methods project the feature vector to a new dimension creating entirely new features.

## 3.5 Feature Normalizations

Machine learning algorithms learn the mapping of the input feature vector to the target class; therefore, the scale and distribution of the features of the input samples maters for high classification performances. One common way to mitigate the effect of the different scaling of the input feature vector is to linearly rescale the input features, which refers to data or feature normalization. ML algorithms that use distances between the input samples, such as kNN and SVM, are affected by the scale

differences in the input data. Other classifiers, such as DTs, optimally split the input data using percentages of the correct classified labels and, therefore, are not sensitive to scale differences.

Division by Range data normalization is one of the standard methods for rescaling the input feature values into a range between 0 and 1 using minimum and maximum values calculated over the whole input data. The value is normalized as follows:

$$\acute{x} = \frac{x - min}{max - min} \quad (3\text{-}17)$$

Where $min$ and $max$ are the minimum and the maximum of the input values, $x$ and $\acute{x}$ are original value and its normalized value.

Standardization or Z-score is another way to rescale the input data using the mean and standard deviation over the training data. An input value x is standardized as:

$$\acute{x} = \frac{x - mean}{sd} \quad (3\text{-}18)$$

Where mean and sd are the average and standard deviation of the input values, $x$ and $\acute{x}$ are original value and its normalized value.

This process results in a set of values with a mean of zero and a standard deviation of one. The input data must have a Gaussian distribution, and a well-behaved mean to get reliable results.

## 3.6 Summary

This chapter presented the background theory and procedures for texture analysis in computer vision. Various steps of typical ML methods are explained, including pre-processing, object detection, texture feature extraction, feature selections, and feature normalizations. Pre-processing techniques such as de-noising, image enhancement, illumination normalization and ROI segmentation are described. Some common feature extraction techniques such as LBP, HOG, and FD and representations of extracted features through statistical moments are presented. Both cracks in the natural images of glasses and concrete and nodule border irregularities from US images are reflected in intensity variations across the image and can be well discriminated by densely sampled texture features like LBP and HOG. On the other hand, border irregularity from the nodule border curve is a case of the curve fractals and hence can be captured by Fractal Dimensions. Further, we have shown that features can be extracted in domains other than spatial domains, such as the frequency domain, which is again used in nodule border irregularity recognition since borderline curves can be presented as distances function fluctuations of different frequencies. Finally, the statistical moments of the histogram of Linearity and curvature indicators are used for crack recognition. The rest of the thesis will be guided by various components of the AI framework pipeline, as discussed in this and the previous chapter. Many of the presented ML

pipeline steps and feature extraction methods are used in our two case studies presented in chapters 4, 5, and 6, and we developed several new methods.

In the next chapter, we present our first case study of abnormality assessment of building material. More specifically, we present several methods for surface crack recognition of the glass and concrete from 2D visual images.

# Chapter 4: Analysis of Abnormal Artifacts in Natural Images

This chapter presents the first case study for abnormality recognition in visual images. More specifically, we aim to automatically recognise abnormal artefacts in natural digital images captured by commercially available cameras, focusing on monitoring and assessing the state of building facades and concrete surfaces for safety purposes. We will consider two cases relating to two different building materials, each with its own challenges but having common characteristics. In particular, we develop several hand-crafted feature-based AI algorithms and a few CNN models and test their effectiveness in identifying cracks on concrete and glass surfaces. For the hand-crafted feature-based algorithm, we shall use some texture features that are known for successful use in general image analysis, but we also introduce new types of texture features that are specific for crack abnormality artefacts in glass and concrete panels. We shall demonstrate that most of the proposed algorithms achieve high levels of recognition accuracies. We first describe the setup of the investigated problems highlighting the main challenges and potential benefits of using AI to deal with these tasks. Then, we review the methods that are currently used to assess general defects in construction materials and cracks in particular. Finally, various proposed AI algorithms for the intended tasks are presented, and their performances are tested using relevant publicly available and newly collected datasets.

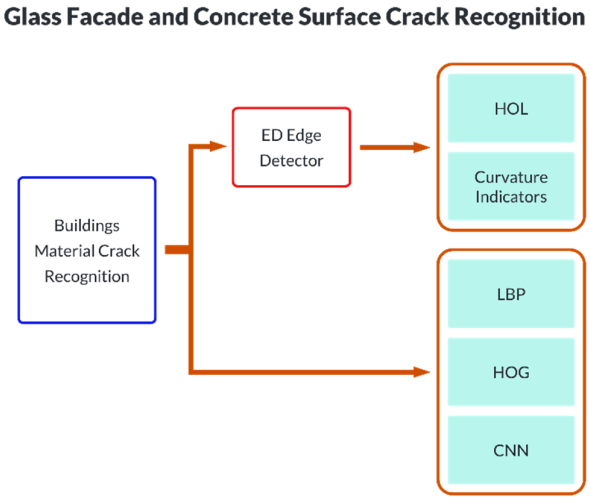



Figure 4.1: Summary of the methods for crack recognition.

## 4.1 Introduction

Most commercial and inner-city sections of densely populated cities worldwide are characterised by high-rise buildings, towers, and skyscrapers (see Figure 4.2a). As explained in chapter 1, the exterior walls and building façades are made increasingly of large, toughened glass panels, which frequently

become defective in the form of cracks. Consequently, they require regular monitoring, maintenance, and renovation to maintain health and safety and ensure energy efficiency and visual appearance. Skilled engineers and staff climb the building with specialised equipment for manual inspections (see Figure 4.2 b). This procedure is time-consuming, costly, and risky. Thus, an automatic inspection using machine-learning-based solutions which can be installed on board Unmanned aerial vehicles (UAVs) is highly desired (see Figure 4.2.c).

(a) (b) (c)

Figure 4.2: (a) Skyscraper, (b) high building worker, (c) drone [126]–[128].

UAVs, including drones, have become more and more popular in recent years for a variety of inspection applications [129]. Vision cameras onboard UAVs can record images in various levels of infrastructure details, such as cracks on the concrete surfaces (see Figures 4.3 and 4.4) and on glass façade.

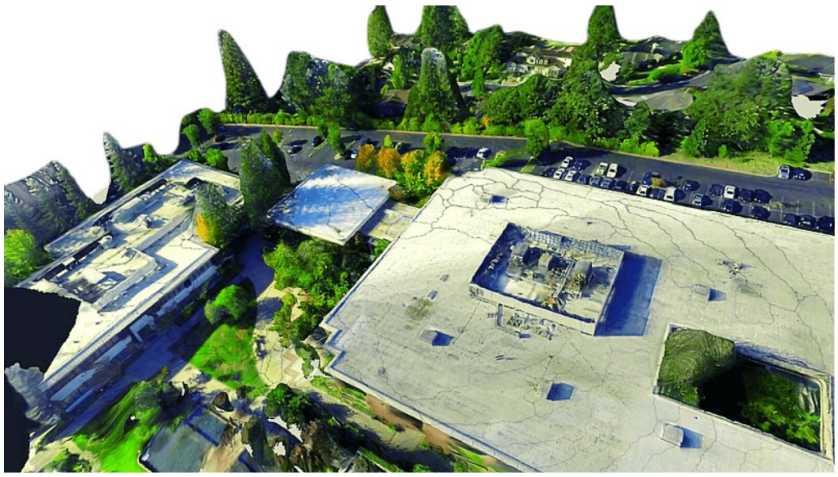


Figure 4.3: Concrete cracks on the top of the building [130].

Figure 4.4: Automatic bridge concrete crack detection using drones [131].

The availability of relatively cheap UAVs endowed with different imaging sensors and cameras provides affordable means to integrate UAVs in experts-led façade assessment and monitoring to deal with the tasks mentioned above as frequently as necessary in a safe and convenient environment. Advances in computer vision algorithms can be exploited to support such tasks in different ways and for different purposes in real-time or offline.

In this thesis, we are concerned with detecting/recognizing cracks in glass panels as well as concrete blocks by automatically analysing natural images of such panels, ideally recorded by UAV cameras. Computer vision systems for such tasks are expected to deal with different challenges, such as reflections, weathering conditions, illumination. when assessing façade glass and concrete surfaces for cracks, although they share similar objectives. Further, the ML algorithms need to deal with different natures of the cracks, for instance, single massive cracks or small crack fractions all over the place (see Figure 4.5).

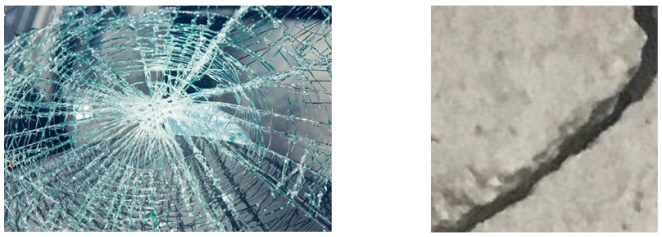


Figure 4.5: Glass cracks with small fractions and single massive concrete crack.

The next section will review existing literature on crack recognition and related works. Before presenting our proposed methods, we introduce our datasets of glass and concrete cracks and briefly describe the necessary image pre-processing. Then, we present several proposed algorithms depending on the specific texture feature used for the intended image analysis, followed by the experimental setup and the results. The chapter concludes with results analyses and recommendations.

## 4.2 A Survey of Glass and Concrete Cracks Techniques

The automatic crack assessment of two building materials, glass and concrete, as well as other similar materials, like solar panels and steel sheets, is the focus of reviewed work in this section. Automatic analysis of contents of the various recorded images/videos aim to extract texture feature vectors/function representing objects and shapes of interest (e.g., edges or other complex shapes) and submit these numerical representations for statistical analysis or other forms of mathematical analysis. A valuable, relevant source of information on image texture analysis is contained in the extensive survey of Liu et al. [65], already described in chapter 2. We need to remember that texture features are either extracted densely (dense sampling) involving each pixel or sparsely (sparse sampling) involving only certain key points in the image. For example, the LBP and HOG feature vectors are extracted densely while the various ULBP groups, or combinations of groups, are extracted post-application of the LBP sparsely. Our investigations for crack abnormality recognition include the use of these dense and sparse texture features in addition to some deep learning-based methods.

There are only a few works done on glass crack recognition, while there are many works on cracks/defects in other building materials like concrete, asphalt, and mortar. In [132], the glass cracks are detected by first pre-processing the input images using Neighbourhood Averaging for noise reduction and the Laplacian filter for sharpening the input image. The suspected cracks are then segmented using adaptive thresholding with subsequent measurements of the area, perimeter, and object roundness. These measurements are used to assess the crack's characteristics; however, neither the experimental dataset nor the method's performance is presented. An automatic detection and recognition algorithm for safety glass crack fragments is proposed in [133]. The glass images are first pre-processed using a Sobel edge detector to generate suspect glass crack edges, followed by binarization using OTSU thresholding. The crack edges are smoothed and connected using morphological operations. Finally, the watershed transformation method based on chain code is used to segment the glass fragments, followed by determining the shape, size, and glass fragment numbers. The method showed effectiveness for crack segmentation and crack parameter calculations. The work is done on very small dataset of glass images under manufacturing conditions where reflections and other artifacts such glass panel frames are not present.

An automatic crack detection method from noisy concrete surfaces is proposed in [134]. As a pre-processing step, a median filter is used to remove shades on the concrete surface, and a multi-scale line filter with the hessian matrix is used to sharpen the crack segments against blebs and stains and to smooth the variation in crack width. The crack detection is done in two steps; (1) the probabilistic relaxation method is used to detect the crack and remove the noises coarsely, and (2) local adaptive thresholding is applied to detect the crack segments more precisely. Sixty noisy concrete images, including various blebs, brains, and shadings, are used to evaluate the method's performance. The

proposed method achieved a specificity and sensitivity of 80% and 99%, respectively. Using the proposed method with the two pre-processing steps showed an area under the curve (AUC) improvement achieving 99% compared with the method without pre-processing steps. An automatic detection method for bridge concrete cracks is proposed in [135]. The median filter for image smoothening and the Canny method for crack edge extraction are used as pre-processing steps. Then, the crack edges are segmented using a modified region-based active contour algorithm. The noises are eliminated using a linear support vector machine based on a greedy search strategy. The final step involved marking each crack segment and estimating its width using the contour tracing algorithm. The methods evaluation on 50 noisy concrete crack images shows an accuracy rate above 92.1% for estimating the mean width of the detected cracks. The HOG-based method is used to detect cracks on concrete surfaces from binary input images [24]. The proposed method showed significant emphasized cracks when tested on two concrete images containing a horizontal and a vertical crack. However, methods evaluation on only two images with two individual cracks is not adequate to judge the effectiveness of the proposed method. The HOG feature is used in our case study of abnormality assessment for both glass and concrete images; however, it is used for crack recognition instead of crack detection.

There is a long list of works done on crack detection and recognition in concrete images using deep learning. We introduce in the following some of the recent works. In [136], the author uses seven CNN architectures based on transfer learning (pre-trained on ImageNet) for concrete crack detection. Several performance analyses, such as the effect of the training dataset's size, the network's depth, the number of training epochs, and the expandability of the method to other building materials, are investigated using different CNN architectures. The proposed methods are evaluated on a publicly available dataset [137] of 40000 image patches generated from 500 high-resolution images collected from buildings at Turkey's Middle East Technical University (METU) campus. The methods achieve accuracies as high as 96%. The same publicly available dataset of concrete cracks is used in our work to evaluate our proposed crack recognition methods. In another work, AlexNet CNN architecture in transfer learning mod is used for concrete crack detection [138]. The high-resolution dataset is split into 1250 images for training and validations and 205 images for testing. The training and validation dataset are cropped into 60000 image batches of an equal number of cracked and non-cracked batches. The average test accuracy on 205 images using a sliding window exhaustive search was 99.09%. The next work uses the CNN model in transfer learning based on improved EfficientNetB0 architecture for concrete crack detection [139]. The model number of parameters is significantly reduced compared with MobileNetV2, DenseNet201, and InceptionV3, giving an accuracy of 99.11%. Reducing the number of parameters in the CNN model makes it lightweight for installation on mobile devices. This point is particularly interesting for the current work since our work's ultimate aim is to install a crack recognition method on board a UAV for automatic building inspection.

## 4.3 Glass and Concrete Datasets

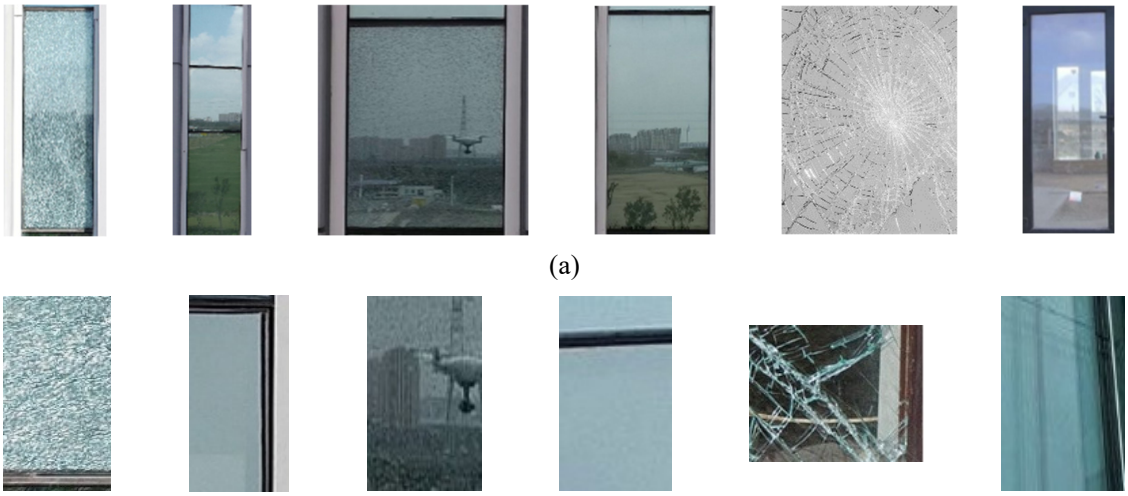
We collected four glass crack datasets using a drone and Google search to evaluate our proposed methods and used one publicly available concrete surface dataset [137]. Two glass datasets are formed from videos taken by a drone from an office building site in the LinGang Business District of Shanghai City, China (see Figure 4.6). Several drone flights were conducted at different daytimes and weather conditions. A DJI Phantom 4 Pro V2.0 drone with 20 megapixels in 4K quality and 60 frames per second onboard camera is used. The 4K camera resolution is used to collect the D4K dataset, while the same camera in HD format is used to acquire the DHD dataset. By choosing these two resolutions, we can evaluate the performance of our proposed methods on low- and high-quality images. The images are manually cropped from the recorded video frames and labelled as cracked and non-cracked.

The videos are recorded on four sides of the office building with seven cracked glass panels. Due to the restricted number of cracked glass panels, two images from each broken panel were sampled from the recorded videos. Thus, balanced datasets are created using a total of 14 images of broken glass panels and an equal number of non-cracked glass panel images. Since the number of the original images is low and their resolution and view angles are large, each of the 14 images of both image classes is split into non-overlapping four-by-four image batches, creating a collection of $14\times16 = 224$ images of each of the glass classes for both D4K and DHD datasets. The third dataset DG is collected using a Google search of initially 68 images from both cracked and non-cracked images of various qualities and sizes. Some of the images had to be manually cropped because the actual glass panels were not in the centre of focus. The images are then divided into four by four non-overlapping batches similar to D4K, and DHD described earlier, producing a total of 2176 image batches. However, as some of the cracked photos are only partially cracked, some of the batches from the original cracked images did not contain the actual crack segments, leading to a highly unbalanced number of cracked and non-cracked image patches. Consequently, the image batches are down sampled to produce the final DG dataset of 896 by 896 images from both classes. Figure 4.7 below shows examples of the original images (see Figures in a) and some of the batches created from the original images (see Figures in b).

Finally, a new dataset, ALL, is created by merging the three datasets to evaluate our proposed methods on a dataset collected from different sources and having different qualities. The dataset is formed with equal portions of 224 images from both image classes from each of the datasets: D4K, DHD, and DG. The only dataset that needed to be down sampled was DG. Therefore, we randomly selected 224 images of both classes from DG and merged them with the other two datasets into the ALL dataset, creating a balanced dataset of $2\times672 = 1{,}344$ image batches of both classes (see Table 4.1).

As an additional data preparation step, all images in individual datasets are resized to the average size of the dataset calculated over both classes of glass images to mitigate the effect of image size variations across the datasets on the performance of the proposed methods.

Figure 4.6: Glass façade of an office building in LinGang District, Shanghai (reflected drone image in the red circle).



(a)

(b)

Figure 4.7: (a) Examples of original cropped images (b) batch images of cracked and non-cracked glasses.

The concrete dataset is publicly available [137] and consists of 40000 image patches of equal sizes of 227x227 pixels and equal amounts of cracked and non-cracked concrete surfaces (see Figure 4.8). The image batches are generated from 458 high-resolution images taken at the buildings of METU in Turkey. The high-resolution images are cropped to build the image patches using the method described in [140]. It is worth mentioning that no augmentations are used to build the dataset, and the images are taken from different concrete surfaces under different illumination conditions.

Figure 4.8: Two non-cracked (on the left) and two cracked (on the right) concrete image patches.

Table 4.1: Shows the datasets used in our experiments and their sizes.

| Dataset Sizes | |
|---|---|
| Datasets | Cracked, Non-cracked |
| D4K | 224 , 224 |
| DHD | 224 , 224 |
| DG | 896 , 896 |
| ALL | 672 , 672 |
| Concrete | 20000 , 20000 |

## 4.4 Proposed Methods

The following sections describe our proposed methods for recognizing cracks in the glass and concrete construction materials. The overall workflow of our proposed methods is presented in Figure 4.9. The methods are based on: (1) extracting crack edges to calculate linearity or curvature indicators to build a histogram, then, either the histogram or its statistical moments are used as a feature vector for input to a classifier, (2) extracting textural features such as ULBP and HOG to build a histogram for input to a classifier. In this sense, we first describe the adopted edge detection algorithm used as a pre-processing step for some of the methods and then present our proposed methods.

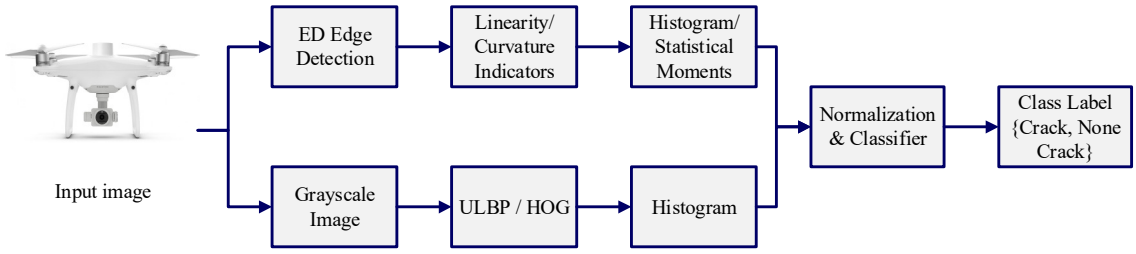



Figure 4.9: An overview of the proposed method's main step for crack recognition.

### 4.4.1 Adopted Pre-processing - The Edge Drawing Algorithm

The nature of cracks in different/similar material surfaces differ depending on the quality of the manufactured material, but all share some typical characteristics. For example, they all appear as curved or line-like objects of different thicknesses. In the survey above, most of the reviewed crack algorithms apply denoising and edge detection procedures besides morphological operation to segment the crack fragments. Having tried many pre-processing methods, we found that using an edge detector to analyse the glass crack edge segments is the most useful for our work. Various well-studied edge detectors, such as Roberts [61], Sobel [49], and Canny [141], are used in many computer vision applications. Here, the recently published edge drawing (ED) [64] algorithm is used as an edge detector for its high performance in addition to producing edge segments of connected pixels and always one pixel wide (see section 3.3.1). These properties of ED specifically simplify the calculation of the linearity and curvature features introduced in the coming sections.

### 4.4.2 Histogram of Linearity

Most objects in natural images are easily recognizable when their edges are extracted. The cracks in the glass panels or concrete surfaces can also be recognized by examining their edges, showing many line-like structures in contrast to other artefacts, such as dirt or reflections from nearby objects. This observation led to considering the Histogram of Linearity (HOL) features to capture glass and concrete cracks. Linearity measures how much a set of points represents a perfect line. Usually, the linearity value "1" is assigned to the shapes having their points lying on a straight line, and "0" is assigned to a totally unordered set of points, such as spots. The linearity feature is already used in many applications, including shape recognition [142]. Correlation is one of the linearity measures which shows how strong the relationship between two variables or a set of points is. The set of points lying on/around a line have the strongest correlation of either -1 or +1. The sign here shows a positive or a negative correlation, i.e., the relation or correlation between the two variables is going in the same or opposite directions. This feature is similar to the concept of HOG in that HOL is built from a histogram of the slopes (correlation coefficients) of the connected components post-edge detection.

To measure the linearity of the glass cracks, we first extract edge segments from the image and then measure the linearity of the individual edge segments. Pearson's linear correlation coefficients (equation 4-1 [143]) are used to measure the linearity.

$$r = \frac{n\sum_{i=1}^{n} xy - \sum_{i=1}^{n} x \,.\, \sum_{i=1}^{n} y}{\sqrt{n\sum_{i=1}^{n} x^2 - \left(\sum_{i=1}^{n} x\right)^2}.\ \sqrt{n\sum_{i=1}^{n} y^2 - \left(\sum_{i=1}^{n} y\right)^2}} \qquad (4\text{-}1)$$

Here, r is the linearity value, n is the number of pixels forming the edge segment, and x and y are the coordinates of the edge segment points. In our method, we used only the absolute value of r, i.e., the sign is ignored since the crack segments' orientation does give any extra indications regarding cracks.

Images of cracked glass panel surfaces should contain more line-like segments with linearity values in a certain range than non-cracked glass images. Hence, a histogram of linearity values can be suggested to represent the difference between cracked and non-cracked images. As shown from the two histograms in Figure 4.10, the frequency of the linearity values closer to one of a cracked glass image is overall higher than that of the non-cracked glass image. It can be further observed that the number of edge segments in the cracked image is higher than that of the non-cracked one, which can be explained by the existence of many small crack fragments in the case of cracked glass. Hence, the histogram of linearities is used as a discrimination feature vector. For each histogram bin, the number of edge segments having bins' specific linearity value range are counted (see Figure 4.10). The histograms are normalized by dividing each bin by the sum of all bins. Feeding the histograms directly to the classifiers may face the issue of high dimensionality in addition to lower discriminability. Statistical moments are simple, of low dimension, and reflect the distribution characteristics already used in many classification

tasks; hence it is used for crack recognition. Five statistical moments mean (μ), standard deviation (σ), entropy (H), skewness ($\mu_3$), and kurtosis ($\mu_4$) are derived from the normalized histogram of linearity and used as a feature vector for the input to the classifier. The description and equations of the individual statistical moments are presented in section 3.4.1. These five features should depict different aspects of the linearity distribution across the histograms and can discriminate the cracked images from non-cracked ones alone and jointly.

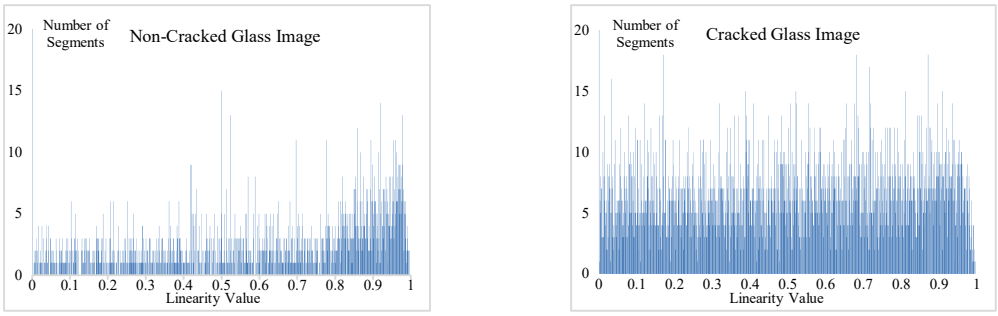



Figure 4.10: Examples of linearity histograms for both cracked and non-cracked glass images.

Further investigations of the edge segments reveal that they have high variations in length, which may affect the linearity values and reduce the method's performance. Therefore, the edge segments are split into equal pieces of different lengths (split size) before calculating linearity, leading to improved method performance.

### 4.4.3 Curvature Indicators

In the following, we present two methods based on curvature and curvature indicators for crack recognition.

#### 4.4.3.1 Histogram of Curvature

Many edge segments shown in different colours (see Figures 4.11 a and b) of cracked and non-cracked glass and concrete images reveal nonlinear structures, hence the method's limitations based on the linearity feature. We can further see in the histograms shown previously (see Figure 4.10) that many edge segments in both image classes have linearity values close to 0, which indicates non-linear structures. In order to capture these curvy-shaped crack segments, a curvature feature measuring the bending strength of the crack segments can be employed. Curvature is defined as the amount by which an object deviates from being a flat surface or a curve from being a line. Positive curvature values represent convex shapes or curves, while negative values represent concave curves. Zero curvature values usually show the crossing points from convex to concave regions and vice versa on the curve or contour of a shape.

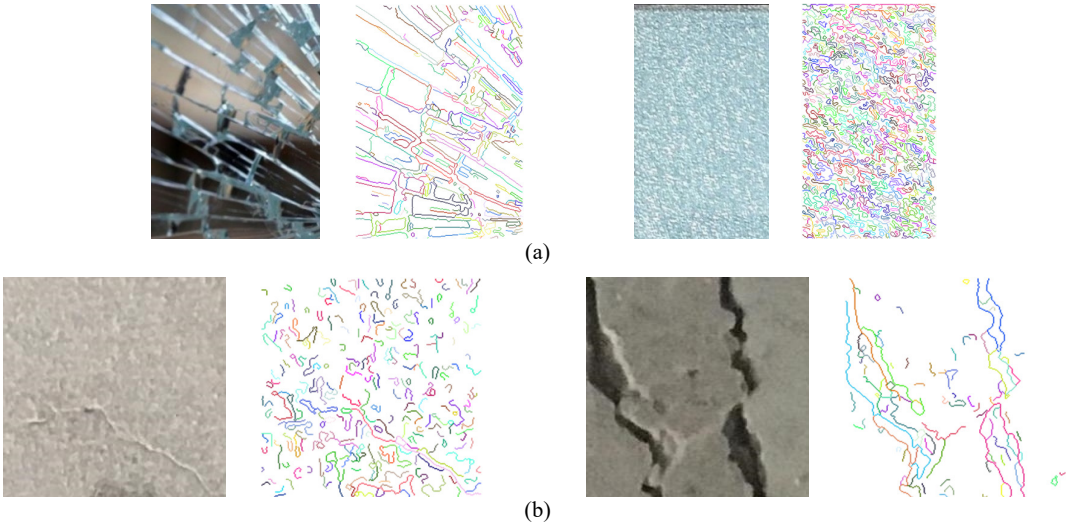


Figure 4.11: ED edge detector output of glass panels (a) and concrete (b) with massive cracks and small crack fragments. The different colours show individual edge segments just for illustrations and have no specific meaning.

Curvature can also be defined as the rate at which a tangent to a curve changes direction [144]. The curvature (κ) can be given by the following equations:

$$T = \frac{dr}{ds} \tag{4-2}$$

$$k = \left|\frac{dT}{ds}\right| \tag{4-3}$$

$$k = \left|\frac{dTdt}{dtds}\right| = \frac{1}{\left|\frac{ds}{dt}\right|}\left|\frac{dT}{dt}\right| = \frac{1}{|v|}\left|\frac{dT}{dt}\right| \tag{4-4}$$

$$where\ T = \frac{v}{|v|}$$

$T$ is the unit tangent vector, s is arc length, and r is the unit vector direction.

By imagining a particle moving along a curve in a two-dimensional surface with its actual position given by x and y coordinates, we represent (s) in a parametric equation form:

$$s = x(t) + y(t) \tag{4-5}$$

The variables x and y are dependent on the third variable, t.

$$v = \frac{ds}{dt} = \frac{dx}{dt} + \frac{dy}{dt} = x' + y' \tag{4-6}$$

$$|v| = \sqrt{x'^2 + y'^2} \quad (4\text{-}7)$$

$$T = \frac{v}{|v|} = \frac{x' + y'}{\sqrt{x'^2 + y'^2}} = \frac{x'}{\sqrt{x'^2 + y'^2}} + \frac{y'}{\sqrt{x'^2 + y'^2}} \quad (4\text{-}8)$$

$$\frac{dT}{dt} = \frac{y'(y'x'' - x'y'')}{(x'^2 + y'^2)^{\frac{3}{2}}} + \frac{x'(x'y'' - y'x'')}{(x'^2 + y'^2)^{\frac{3}{2}}} \quad (4\text{-}9)$$

$$\left|\frac{dT}{dt}\right| = \sqrt{\left[\frac{y'(y'x'' - x'y'')}{(x'^2 + y'^2)^{\frac{3}{2}}}\right]^2 + \left[\frac{x'(x'y'' - y'x'')}{(x'^2 + y'^2)^{\frac{3}{2}}}\right]^2} = \frac{|y'x'' - x'y''|}{|x'^2 + y'^2|} \quad (4\text{-}10)$$

$$k = \frac{1}{|v|}\left|\frac{dT}{dt}\right| = \frac{1}{\sqrt{x'^2 + y'^2}} \frac{|y'x'' - x'y''|}{|x'^2 + y'^2|} = \frac{|y'x'' - x'y''|}{(x'^2 + y'^2)^{\frac{3}{2}}} \quad (4\text{-}11)$$

Where x and y are the edge point coordinates on the image for which we are calculating the curvature. $(x', x'', y', y'')$ are the first and second derivatives of x and y, respectively [144]. In digital image analysis applications, we need to use the following discrete versions of derivatives:

$$f'_x(t) = \frac{f(t+n)_x - f(t)_x}{n} \quad (4\text{-}12)$$

$$f''_x(t) = \frac{f(t+2n)_x + 2 * f(t+n)_x - f(t)_x}{n} \quad (4\text{-}13)$$

Here, $f'_x(t)$ is the first derivative for a continuous 1-dimensional function $f(t)_x$ with respect to t, $f''_x(t)$ is the second derivative for a continuous 1-dimensional function $f(t)_x$, and $n$ is the distance in pixels between two consecutive points (here referred to as sample size).

As shown in equations 4-12 and 4-13, the curvature can be calculated at any point along the detected edge using first and second derivates. Three consecutive points along the edge are needed for these calculations. The edges are extracted using the ED edge detector [64]. Using different values of n captures different geometrical aspects of the edge segments, and the best-performing value is determined empirically during the experiments. The curvature values calculated at the segment points are then accumulated in a histogram before feeding to a classifier.

#### 4.4.3.2 Connected Pixel Configurations

Extensive experiments show that calculating curvature using step size n=1 performs the best for crack recognition. Thus, the curvature calculated for every three connected pixels always yields a certain value depending on the pixel's geometrical alignments. This fact inspired us to use three connected pixel configurations as an indicator for edge curvature. The histogram of connected pixel configurations is used for crack recognition instead of the histogram of curvature, yielding better performances and

simpler to calculate. Connected pixel configurations can be explained by taking three neighbouring pixels (connected) along the edge segment; then, we calculate how many different shapes these connected pixels can take and build statistics for discrimination of the cracked images. The 3-connected pixel configurations (3PixelConf) can take 64 different shapes or configurations, given that they are always connected. Similarly, the 4 and 5 connected pixel configurations can take 512 and 4096 possible configurations. As a reminder, the different pixel configurations show how the neighbouring pixels are geometrically aligned to each other and therefore indicate geometrical structures of the edge segments similar to the curvature. Figure 4.12 shows some of the 3PixelConf numbered from 1 to 64. A histogram with a number of bins equal to the number of possible configurations (here 64) is calculated. The bin frequency represents the number of occurrences for each configuration across all edge segments of an image. Again, using an ED edge detector to extract edges make it easier to calculate pixel configurations for its property of one pixel wide and edge segments of connected pixels. The more pixels (e.g. 3, 4, 5) are considered for the connected pixel configurations, the higher the number of possible configurations and the higher the computation demands. Therefore, smaller numbers of pixels should be considered, especially for installation on mobile devices. We attempted the 3, 4, and 5 connected pixel configurations in our experiments; however, three connected pixel configurations gave slightly higher performances. Hence we use only 3PixelConf in our experiments.

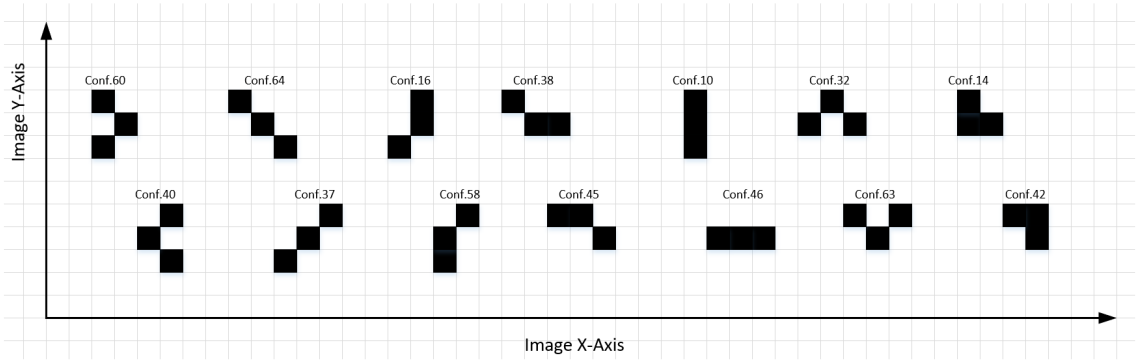



Figure 4.12: Some of the 3-connected pixel configurations.

Figure 4.13 below shows the histograms of 3PixelConf configurations calculated from the cracked and non-cracked glass images over the D4K dataset. The histograms reveal, among others, that the frequency of the different configurations of the cracked images is higher than that of the non-cracked images due to the existence of a higher number of edge segments. Another insight of the histogram shows the high frequency of two-pixel configuration numbers 10 and 46 compared to the remaining ones in both cracked and non-cracked images. These two-pixel configurations correspond to the long horizontal and vertical line edges produced from the long frames of the glass panels in our datasets. In addition, the high number of the two configurations could reflect the existence of line-like edge segments from, for instance, single massive cracks, especially in the concrete dataset. Further investigations are needed to determine the discriminative power of these two configurations.

The histogram also reveals many pixel configurations (e.g., Conf 2, 3, 5, 42, 43) having a frequency of 0 and might be removed to reduce the dimensionality of the feature vector without affecting classification performance. It is also interesting to know why some connected pixel configurations are completely absent among all images of both classes. This must be related to the nature of glass and concrete images and how the ED algorithm extracts the edge segments.

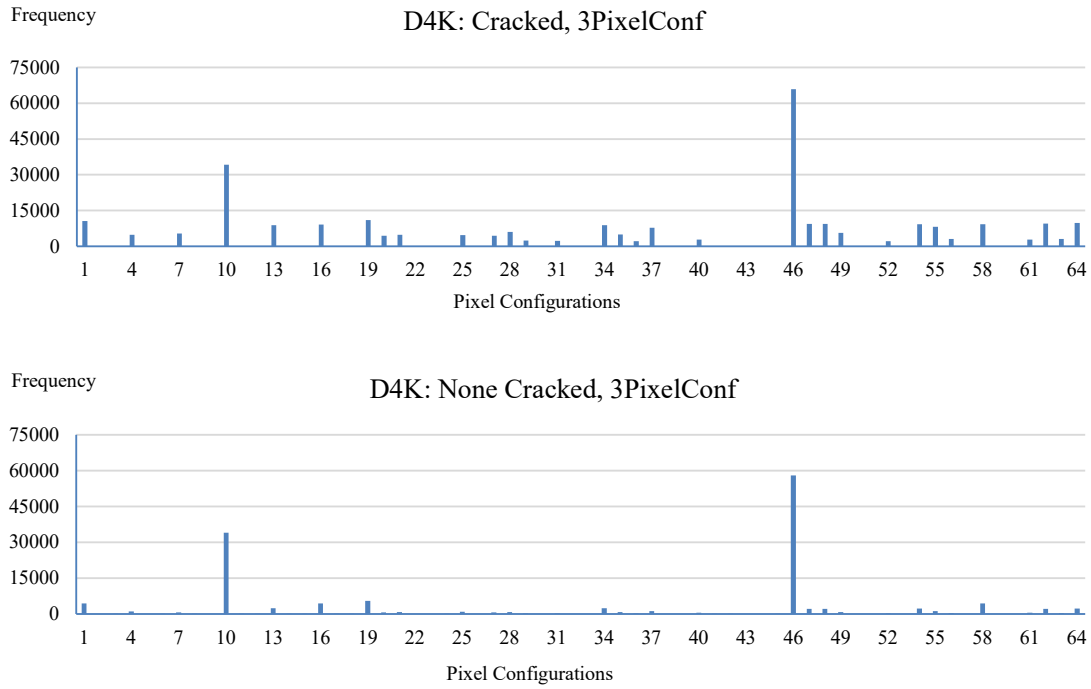



Figure 4.13: Frequency of each of the 64 3PixelConf configurations.

Finally, we introduce the two possible overlapping and non-overlapping approaches for calculating connected pixel configurations, as shown in Figure 4.14 below. In the overlapping approach (Figure a), each pixel is considered part of at least one and a maximum of three consecutive pixel configurations. In the non-overlapping approach (Figure b), a pixel can only be part of one configuration. Consequently, the frequency of the configurations in the overlapping approach is higher than in the non-overlapping approach. Although the two approaches performed similarly, the non-overlapping approach may be preferred due to its lower computational costs.

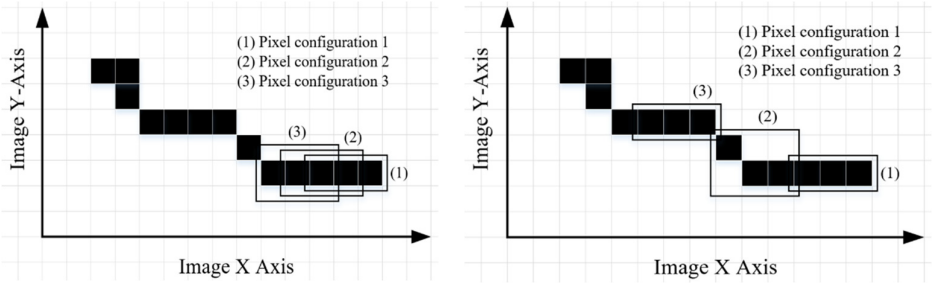



Figure 4.14: (a) Overlapped pixel configurations (b) None overlapped pixel configurations.

### 4.4.4 The ULBP Method

In chapter 3 (section 3.4.2), we detailed the description of the Local Binary Pattern (LBP) [21] as a texture feature successfully applied in many pattern recognition and computer vision tasks. Accordingly, we first investigate the use of different representations of the LBP texture features for automatic crack recognition schemes. The main incentive for using LBP in our work is its repeatedly demonstrated discriminative power for image texture analysis. The strength is its fast calculation in addition to its invariance to rotation and illumination changes. According to initial experiments on the prevalence of the various types of LBP codes in the glass and concrete datasets, the ULBP counts for around 82-90% of all LBP codes; therefore, it is natural to attempt ULBP with its similar discrimination as LBP and much lower computation demands. This texture feature can be extracted in two different ways: extracted from the entire image region or from the set of pixels on detected edges, and in both approaches, these methods are represented by histograms of different bin numbers depending on the groups of LBP landmarks.

Figure 4.15 shows some examples of LBP representations of glass panels, which show visual differences in LBP texture features between the cracked and non-cracked LBP transformed images.


Figure 4.15: (a) original cracked panel, (b) LBP transformed image of the cracked image, (c) original none cracked panel, (d) LBP transformed image of none cracked panel.

### 4.4.5 The Histograms of Oriented Gradients

As described in section 3.4.3, HOG is a feature descriptor extracted densely at each pixel used originally for pedestrian detection [72]. Unlike LBP, HOG uses local intensity gradients and edge directions for discriminating objects in visual images. Thus, HOG captures the directionality of texture features and might discriminate the crack edges in their directionalities. We implemented HOG slightly different than the original paper on pedestrian detection. The whole input image is divided into three by three equal rectangular cells, and then the gradient magnitude and edge directions are calculated at each pixel. A histogram of 9 bins (180°/9 = 20°) of the gradient magnitudes and directions (or edge orientations) is formed for each cell. The cell histograms are concatenated to build a histogram of a total of 9X9=81 bins. The pre-processing step of contrast normalization of the local intensities is not attempted in opposition to the original paper. Finally, the histogram (81 bins) is normalized (by dividing each bin by the sum of all bins) as in other methods and fed to a classifier.

### 4.4.6 Partitioning-Based histogram of Linearity

The method described here is a modification of the linearity-based method described in the previous section, 4.4.2. Inspired by HOG, we partition the input greyscale image into (3 by 3) equal blocks before extracting the linearity feature. The experimental results show that partitioning the input image into equal blocks improves the accuracy due to the increased dimensionality of the feature vector. We apply the same procedure as before for extracting linearity features and building histograms for individual blocks and apply different split sizes of the edge segments before calculating linearity. The nine individual block histograms are concatenated into 9x1000 bins for use as a feature vector for input to a classifier. Figure 4.16 below shows the partitioned ED output of cracked and non-cracked glass and concrete images.

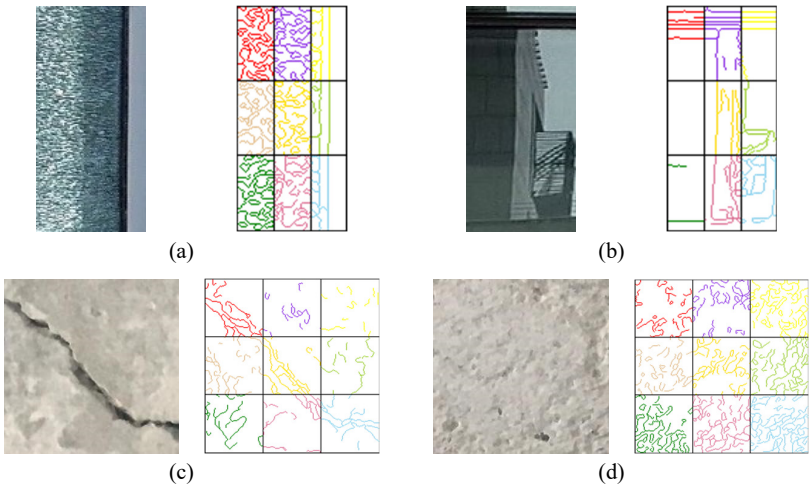



Figure 4.16: Original and ED segments divided into nine equal boxes (a) cracked glass, (b) non-cracked glass, (c) cracked concrete and (d) non-cracked concrete.

### 4.4.7 Partition-based Connected Pixel Configurations (Curvature)

Similar to the previous method, we partition the input grayscale image into nine equal blocks before extracting the ED edges and calculating the histogram of the connected pixel configuration for each block. Again, we apply the same procedure as before for calculating the 3PixelConf features and building histograms for each block. Then, through concatenation of the individual partition histograms, a histogram of a 9x64 dimensional feature vector is formed for the input to classifiers. The division of ED edge detector output into nine boxes and calculating 3Pixelconf from each block is illustrated in Figure 4.17.

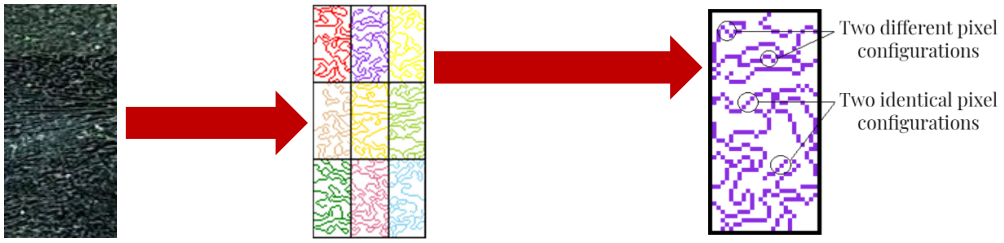



Figure 4.17: Division of the ED output of cracked glass panel into nine boxes and 3PixelConf calculations.

### 4.4.8 Partition-based ULBP Method

Like the previous two methods, we apply the partitioning of the input image or the ULBP code map of the image into nine equal blocks prior to calculating ULBP histograms. Unlike other methods, we partition the ULBP code map instead of the input grayscale image for easier generation of the ULBP codes at the individual box edges. The block histograms of 58 bins are concatenated to build a feature vector of 9x58 bins and fed to a classifier.

### 4.4.9 Deep Convolutional Network CNN based on Transfer Learning

In the following, we describe our CNN models based on transfer learning. Deep learning (CNN) models, described previously in section 2.3, have recently gained much attention for their high performance in many classification and recognition tasks. Usually, training a CNN model from scratch requires a large dataset; however, using transfer learning techniques gives high performance on relatively small datasets. In transfer learning, the CNN model is pre-trained on a big dataset, usually of natural images such as ImageNet [37], and then used to classify the smaller dataset of the new task using transfer learning techniques. The pre-trained CNN model already learned general features such as edges, colours, and texture patterns and can be reused for feature extraction on the new dataset. Two CNN architectures, VGG16 [145] and ResNet [146], based on transfer learning with the same architectural parameter settings, are used for glass and concrete crack recognition. The choice of using VGG16 and ResNet50 is to evaluate CNN architectures from two different families. The VGG16 has a sequential architecture, while ResNet50 is a member of the ResNet architecture family using connection shortcuts (skip connection) to allow very deep convolution layers. The deeper the convolution layers, the higher

is training error, making it unuseful for very deep layers in the CNN architectures. The ResNet family solves this problem by introducing a deep residual learning framework that uses skip connection between the layers to allow low error rates in very deep convolutional layers, as in the case of ResNet50 architecture with its 50 layers. VGG16, on the other hand, is one of the very successful architectures which wins the ImageNet Challenge 2014 and has 16 layers and small convolution filters of 3x3.

The two architectures are trained to classify 1000 different natural images; however, we modify the last fully convolutional layer to classify two image classes. As described in section 2.3, there are two transfer learning approaches: transfer learning by feature extraction and transfer learning by fine-tuning. Instead of computationally expensive finetuning of the whole network on the new dataset, we used transfer learning by feature extraction. In this type of transfer learning, the existing pre-trained convolutional layers are used for extracting the features from our training and evaluation datasets and used to train new fully connected layers designed to classify two image classes. Since the models were trained on natural images (ImageNet), transfer learning by feature extraction may perform well when applied to natural images of concrete and glass. However, it might need other considerations for other image types, such as US images. The new fully connected layers include a flattened layer (25088 features), a dense layer (256 outputs), a dropout layer of 50%, and one binary output as the last layer. The architecture of our CNN-based models is shown in the following block diagram (see Figure 4.18) with the illustration of the new fully connected layers:

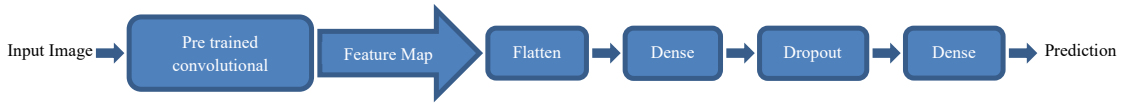



Figure 4.18: CNN-based method showing the fully connected model layers.

## 4.5 Results and Evaluations for Glass and Concrete Images

In this section, we first describe our experimental setup and then present and discuss the results of our proposed methods.

### 4.5.1 Experimental Setup

A five-fold cross-validations protocol is used to evaluate all hand-crafted feature methods, while a one-fold validation is used for the CNN-based models. For the five-fold cross-validation in each of the folds, 80% of the data is used for training and the remaining 20% for testing. Depending on the experiment, the accuracy (**Acc**), sensitivity (true positive rate) (**Cracked**), and specificity (true negative rate) (**Non-Cracked**) are recorded for each of the folds. Then, the averages over the five folds' accuracy, sensitivity, and specificity are calculated. The kNN classifier with k=10 is used in the handcrafted feature-based methods; even though different values of k are tried, the k value of 10 produces better overall results.

An SVM classifier gave slightly lower overall accuracies; hence, only the results of using kNN are presented throughout the experiments for consistency.

As a computer, 64 bits windows PC with 16GB RAM and Intel Xeon W-2123 CPU are used to run the experiments on hand-crafted features. Using the same computer machine to train the CNN models in five-fold cross-validations proved inefficient, especially on the concrete dataset with a large dataset consisting of 40000 images. Therefore, the experiments on the CNN models are restricted to a more powerful server using only one-fold training and validations. However, the one-fold training and validation are repeated twice using two different 80% to 20% and 60% to 40% training to testing proportions to see the effect of the training size on the performance. The server machine is a 64 bits Windows PC with 64GB RAM, Intel (R) Xeon E5-2670 CPU, and two NVIDIA GeForce RTX 2080 GPUs.

### 4.5.2 Histogram of Linearity Performance Measure

The five statistical moments described previously in section 3.4.1 are extracted from the histogram of linearity features HOL to classify the cracked and non-cracked images. These moments are not entirely independent; therefore, it is within this research's interest to determine each moment's and its combination's performance. The division by range normalization (section 3.5) is used to mitigate the differences in the moment's scale before training the classifier.

Table 4.2 summarizes the average overall accuracies (five-fold cross-validations) for each of the five moments calculated from the histogram of 1000 bins of Linearity values. We attempted splitting the edge segments using different split sizes before calculating linearity, where split size =4 performed better, hence the use of this split size in all experiments presented below. The five moments performed equally well, and none of the moments can achieve high levels of accuracy alone. However, the entropy $H$ and the standard deviation $\sigma$ perform slightly better than the remaining three moments due to the reflection of the more uniform distribution nature of the linearity values.

Table 4.2: Linearity feature classification using statistical moments. split size=4

| Datasets | Statistical Moments for Linearity | | | | |
|---|---|---|---|---|---|
| | $\mu$ | $\sigma$ | $\mu_3$ | $\mu_4$ | $H$ |
| D4K | 77 | 81 | 79 | 79 | 81 |
| DHD | 70 | 71 | 70 | 68 | 73 |
| DG | 84 | 82 | 82 | 83 | 82 |
| ALL | 78 | 80 | 77 | 76 | 81 |

The average accuracies using some of the combinations of moment pairs and all five moments on the four datasets are presented in Table 4.3. Combining pairs of moments improves the accuracy marginally, and combining all five moments does not significantly improve the accuracy. The mean and standard deviation combinations perform slightly better than the others and almost the same as all five moments together, showing that they complement each other better. However, there are different

levels of accuracy among different datasets, which may reflect the image quality. The accuracy of the high-resolution datasets D4K and DG is higher than that of the low-resolution dataset DHD.

Table 4.3: Accuracy of some combinations of the linearity moments. Split size=4.

| Datasets | Pair of Statistical Moments | | | | |
|---|---|---|---|---|---|
| | (μ & $\sigma$) | (H & $\sigma$) | ($\mu_3$ & $\mu_4$) | (H & $\mu_3$) | (μ & $\sigma$ & H & $\mu_3$ & $\mu_4$) |
| D4K | 81 | 81 | 79 | 82 | 82 |
| DHD | 76 | 75 | 69 | 74 | 75 |
| DG | 86 | 83 | 83 | 83 | 85 |
| ALL | 83 | 82 | 78 | 83 | 84 |

The edge segments produced by the ED edge detector have different lengths; therefore, splitting the segments into equal pieces may impact the method's performance and hence requires more experimental analyses. For this reason, an experiment is conducted using a range of split sizes starting from 3, as can be seen in Table 4.4. The results reveal slightly higher accuracies when smaller step sizes around 4 or 5 are selected; however, there are no large drops in the accuracy when step sizes increase significantly. Only results for one combination of moments are presented, whereas conducting the same experiments on other combinations gave similar results. It is interesting to mention that the variation of accuracies for lower-resolution datasets appears higher than that for high-resolution datasets.

Table 4.4: Sensitivity analysis of split sizes to classification accuracies (%) for linearity-based features.

| Datasets | Split size ($H$ & σ) | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 15 | 20 | 25 | 50 | 100 | 200 | 1000 | 2000 |
| D4K | 79 | **81** | 79 | 81 | 79 | 82 | 81 | 82 | 81 | 81 | 80 | 84 | 83 | 81 | 80 | 80 |
| DHD | 69 | **75** | 78 | 76 | 75 | 73 | 72 | 68 | 67 | 68 | 67 | 67 | 71 | 69 | 69 | 69 |
| DG | 71 | **83** | 83 | 83 | 84 | 81 | 83 | 82 | 77 | 75 | 74 | 72 | 74 | 75 | 74 | 74 |
| ALL | 77 | **82** | 83 | 82 | 83 | 83 | 83 | 81 | 76 | 74 | 76 | 77 | 78 | 76 | 77 | 77 |

In the experiment presented in Table 4.5, we show the effect of partitioning the input image into nine equal boxes on the method's performance. In contrast to the previous experiments, we present the sensitivity (**Cracked**) and specificity (**Non-Cracked**) in addition to the accuracy (**Acc**). Further, the experiment includes the concrete dataset due to expanding the project in a later stage to the concrete crack recognition. For comparison, we include the experiment results of non-partitioning-based methods in the tables. The results demonstrate that partitioning the input image improves the performance of the linearity-based technique in all datasets of various significance. It shows further that the best performance of the new partition-based approach may appear at different split sizes. The D4K and DHD datasets achieved the best accuracy improvements, around 3% and 4%, respectively, using step size 3. The improvements on the other two datasets were marginal. However, a significant accuracy improvement of around 15% was achieved on the concrete dataset using split size 4. This high accuracy improvement may reflect the existence of massive linear cracks, producing long ED edge segments containing more linear structures when split into pieces, as seen in Figure 4.19.

Table 4.5: Results of non-partitioned and partitioned Linearity-based method.

| Linearity-based Method | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Non-partitioned (split size=4) | | | Partitioned (split size=3) | | | Partitioned (split size=4) | | |
| Datasets | Acc | Cracked | Non-Cracked | Acc | Cracked | Non-Cracked | Acc | Cracked | Non-Cracked |
| D4K | 81 | 82 | 77 | 84 | 81 | 83 | 83 | 82 | 81 |
| DHD | 75 | 72 | 77 | 79 | 72 | 82 | 76 | 68 | 81 |
| DG | 83 | 85 | 79 | 84 | 88 | 81 | 85 | 88 | 82 |
| ALL | 82 | 83 | 82 | 84 | 86 | 81 | 85 | 85 | 84 |
| Concrete | 77 | 79 | 76 | 82 | 72 | 93 | 92 | 92 | 93 |

Figure 4.19: Concrete input image with massive crack and its ED output. Different edge segments are shown in different colours.

### 4.5.3 Connected Pixel Configurations

The first experiment on the 3PixelConf method examines the discrimination power of the individual connected pixel configurations. Figure 4.20 shows the level of accuracy of all 64 connected pixel configurations for each of the four datasets. The results show a substantial number of configurations with 50% accuracy, meaning that they do not have sufficient discriminative powers, while others do. Following that, we examine the impact of deleting the configurations that contribute the least to the classification from the feature vector (see Table 4.6). The results indicate that the average classification accuracy can be as high as 87% when all configurations are combined into a single feature vector. However, the same level of accuracy can be maintained when the less efficient individual configurations are removed from the feature vector. For instance, we can see that even by removing the configurations with accuracies lower than 55% from the feature vector, the remaining 32 configurations can still uphold the same level of accuracy. It can be further observed that although the feature vector dimensionality is reduced by 80% when only 13 best-performing configurations remain, the accuracy only deteriorates marginally. However, the overall accuracy decreases significantly when the number of configurations is reduced to three. The experiment results shown in Table 4.6 are done using the overlapping approach, while Table 4.7 shows the results of the same experiment using the non-overlapping approach. Comparing the two results shows a similar trend of accuracy degradations.

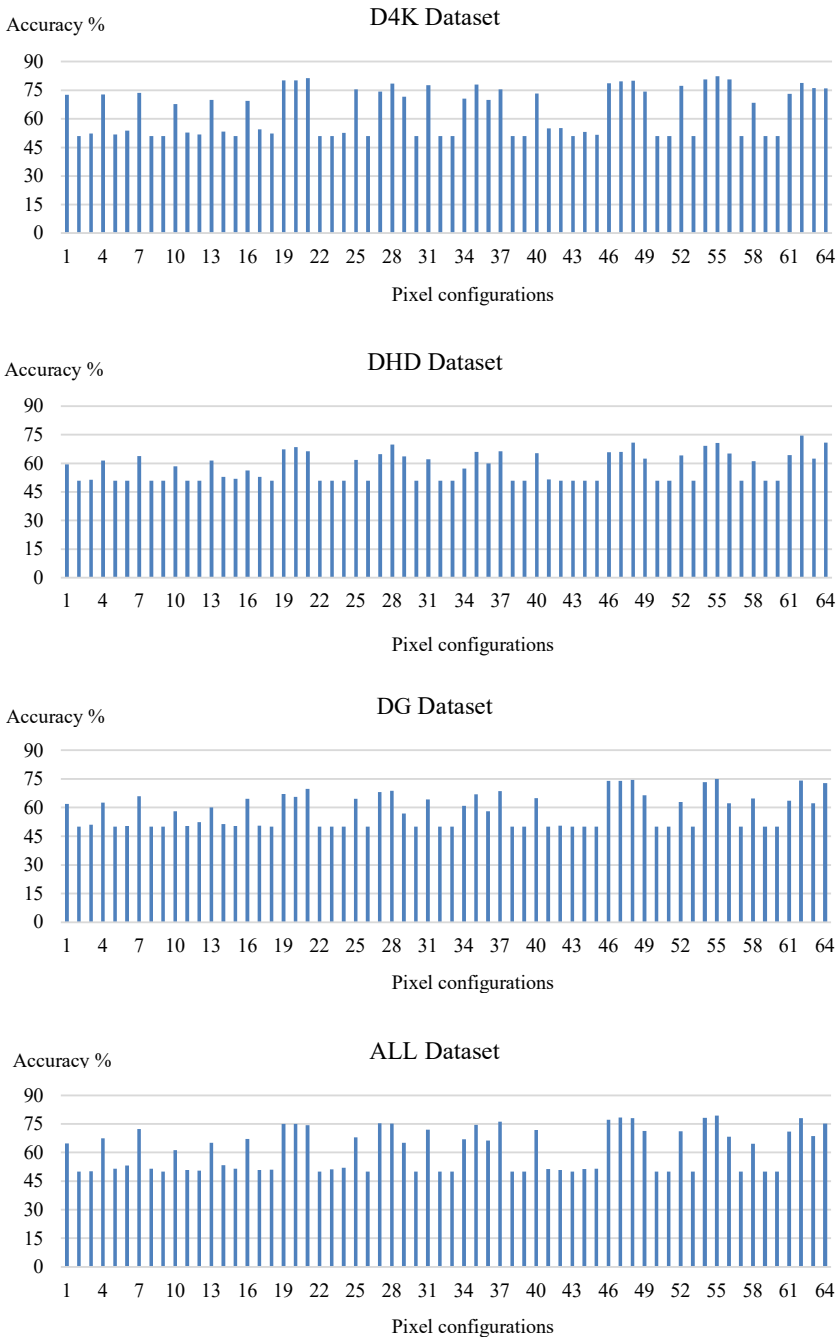



Figure 4.20: Accuracy of each overlapped connected pixel configuration over different datasets.

Table 4.6: Classification accuracies (%) degradation when poorly performed pixel configurations are dropped from the feature vector, bin probability normalized, overlapping approach for pixel configuration used.

| Datasets | Pixel configurations held in Feature Vector with individual accuracies | | | | | | |
|---|---|---|---|---|---|---|---|
| | ≥0% Bins:64 | ≥55% Bins:32 | ≥60% Bins:32 | ≥63% Bins:29 | ≥65% Bins:24 | ≥70% Bins:13 | ≥75% Bins:3 |
| D4K | **85** | **85** | **85** | **85** | 81 | 81 | 75 |
| DHD | **74** | **74** | **74** | **72** | 70 | 72 | 69 |
| DG | **87** | **87** | **87** | **84** | 83 | 83 | 74 |
| ALL | **84** | **84** | **84** | **82** | 79 | 79 | 71 |

Table 4.7: Classification accuracies (%) degradation when poorly performed pixel configurations are dropped, Bin probability normalized, non-overlapping approach for pixel configuration used.

| Datasets | Pixel configurations held in Feature Vector with individual accuracies | | | | | |
|---|---|---|---|---|---|---|
| | ≥0% Bins:64 | ≥55% Bins:32 | ≥60% Bins:23 | ≥63% Bins:16 | ≥65% Bins:14 | ≥70% Bins:1 |
| D4K | **84** | **84** | 81 | 80 | 80 | 50 |
| DHD | **74** | **73** | 70 | 72 | 72 | 50 |
| DG | **87** | **87** | 83 | 83 | 83 | 50 |
| ALL | **84** | **84** | 79 | 79 | 79 | 50 |

## Fusion of Different Configurations

In the previous experiment, we used all 64-pixel configurations and a sub-set of them to build our feature vector for input into a classifier. Here, we attempt to use individual pixel configurations for classification and combine their predictions using the simple majority rule of decision fusion. An odd number of best-performing configurations are selected for the fusion. The results in Table 4.8 below show slight variations in the accuracy when different configurations are fused, with the top three configurations giving the best overall accuracy. The accuracies on different datasets are lower than when pixel configuration was fused at the feature level presented in Tables 4.6 and 4.7 earlier. The non-overlapping approach of calculating pixel configurations (not presented here) gave similar but slightly poorer results.

Table 4.8: Decision level fusion accuracies (%) using a different number of top-performing overlapped connected 3-pixel configurations (Not normalized).

| **Datasets** | **3PixelConf configurations fused at the decision level** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | **3** | 5 | 7 | 9 | 11 | 13 | 15 | 17 | 19 | 21 | 23 | 25 | 27 | 29 | 31 |
| D4K | 81 | **80** | 79 | 79 | 79 | 79 | 80 | 78 | 78 | 77 | 77 | 77 | 75 | 75 | 74 | 74 |
| DHD | 69 | **72** | 71 | 70 | 70 | 69 | 69 | 68 | 67 | 67 | 67 | 65 | 65 | 65 | 64 | 63 |
| DG | 75 | **75** | 75 | 75 | 75 | 73 | 73 | 72 | 71 | 71 | 70 | 69 | 69 | 68 | 68 | 68 |
| ALL | 79 | **78** | 78 | 78 | 78 | 77 | 77 | 76 | 75 | 75 | 75 | 75 | 74 | 73 | 73 | 73 |

As in the previous linearity-based method, we experimented with partitioning the input image into nine equal blocks and calculated the histogram of all 64 3PixelConf configurations for each block separately. Then a histogram of 9x64 bins is built by concatenation and fed to the classifier after normalization. Again, the results of the previous and the new approach of the connected pixel configurations method are presented for comparison, including results on the concrete dataset (see Table 4.9).

Table 4.9: Results of non-partitioned and partitioned-based work using 3-connected pixel configurations-based method.

| | **3PixelConf method** | | | | | |
|---|---|---|---|---|---|---|
| | Non-partitioned (Whole Image - 64 bins) | | | Partitioned (3x3 image blocks - 9x64 bins) | | |
| Datasets | Acc | Cracked | Non-Cracked | Acc | Cracked | Non-Cracked |
| D4K | 84 | 84 | 79 | 88 | 84 | 89 |
| DHD | 75 | 79 | 67 | 78 | 76 | 77 |
| DG | 87 | 86 | 87 | 87 | 89 | 86 |
| ALL | 84 | 88 | 79 | 87 | 85 | 87 |
| Concrete | 79 | 90 | 68 | 66 | 55 | 77 |

An accuracy improvement of about 3-4% was achieved on the datasets D4K, DHD, and ALL; no improvements could be observed using the DG dataset. Surprisingly, there is a significant drop of 13% in the accuracy of the concrete dataset. This can be expected since the massive line-like cracks in the concrete datasets are beneficial for the linearity-based method, while it is not for curvature-like features such as 3PixelConf.

### 4.5.4 Performance of ULBP Feature

We investigate the effect of using different groups of LBP codes on the performance of the LBP-based method by using the basic LBPs (256 bins), ULBPs (58 bins), and ULBP (56 bins) in the experiments (see Table 4.10). The results show no significant differences between the method's overall accuracy for the three LBP code groups. For instance, the ULBP (56) could uphold the same performance as the other two feature vectors but with lower feature dimensionality making it lightweight for installation on mobile devices. The results confirmed the higher efficiency of the ULBP histograms compared with the basic LBP and the hypothesis that ULBP codes 0 and 255 do not contribute much to the discrimination of the cracks. In fact, the two ULBP codes represent dark and bright spots in the image and have no significant relevance to crack lines or curves.

Table 4.10: LBP Classification accuracy (%).

| Datasets | LBP | ULBP (58) | ULBP (56) (0 & 255 excluded) |
|---|---|---|---|
| D4K | 91 | 91 | 91 |
| DHD | 91 | 91 | 91 |
| DG | 88 | 88 | 88 |
| ALL | 88 | 87 | 87 |

Discussing the effect of the LBP radius (see section 3.4.2) applied to different LBP code groups used here is worthwhile. The LBP radius is simply the distance between the centre pixel for which the LBP code is calculated and its neighbour pixels with which it is compared. All previous experiments use the default radius 1, but in principle, any positive integer radius can be used. After conducting a few experiments using a radius bigger than one, which is not presented in this work, it has been established that the default radius (i.e., more local detailed texture patterns) yields the best level of accuracy.

Like other hand-crafted feature methods, we apply the image partitioning of the input image into equal blocks of (3 by 3) on the ULBP-based method. Table 4.11 below shows the results of both partitioning and non-partitioning ULBP methods. There is an increase in accuracy on all datasets when partitioning is applied, except for the DHD dataset. The improvements are marginal for the DG and ALL and around 3.4% and 5% for the D4K and concrete datasets.

Table 4.11: Experimental results of partitioning and non-partitioning based ULBP methods.

| ULBP-based Method | | | | | | |
|---|---|---|---|---|---|---|
| | Non-partitioned (Whole Image -58 bins) | | | Partitioned (3x3 image blocks - 9x58 bins) | | |
| Datasets | Acc | Cracked | Non-Cracked | Acc | Cracked | Non-Cracked |
| D4K | 91.6 | 93.6 | 89.6 | 95.0 | 98.6 | 91.36 |
| DHD | 91.8 | 95 | 88.6 | 91.2 | 94.1 | 88.64 |
| DG | 89.2 | 93.9 | 84.7 | 89.4 | 94.9 | 83.91 |
| ALL | 88.2 | 91.6 | 84.5 | 89.2 | 91.9 | 86.27 |
| Concrete | 91.2 | 90.8 | 92.1 | 96.2 | 94.7 | 97.7 |

### 4.5.5 The Performance of the HOG-based Method

As explained in section 4.4.5, we use (3 by 3) equal rectangular cells to partition the input image prior to extracting HOG to generate a histogram of 9x9=81 bins from the concatenation of the cell histograms. The results of the five-fold cross-validation experiments on all datasets are presented in Table 4.12 below.

Further observations of the results show that the method's overall accuracy on the glass is only outperformed by the partition-based ULBP method. The method's performance on the concrete dataset reaches the highest accuracy of 98%, outperforming all other hand-crafted features.

Table 4.12: Results of HOG-based method.

| HOG Method | | | |
|---|---|---|---|
| Datasets | Acc | Cracked | Non-Cracked |
| D4K | 87.2 | 96.8 | 77.3 |
| DHD | 88.4 | 92.3 | 84.1 |
| DG | 89.0 | 91.1 | 87.3 |
| ALL | 87.2 | 91.5 | 82.8 |
| Concrete | 98.6 | 98.2 | 99.1 |

### 4.5.6 Performance of the CNN Methods based on Transfer Learning

In the following, the experimental results of the two different CNN architectures based on transfer learning by feature extraction for crack recognition are presented. Here, we are interested in knowing how hand-crafted features compete against state-of-the-art AI techniques such as CNN. As mentioned in section 4.4.9, the pre-trained convolutional layers are only used for feature extraction, while the new fully connected layers are used for classification. The architectures are pre-trained on ImageNet, and the new fully connected layers are trained on our glass and concrete datasets for crack and non-crack classes. We use the more powerful computer with 2 GPUs on board to train and test these models using one-fold instead of five-fold cross-validations. Tables 4.13 and 4.14 below show the experimental results of the VGG16 and ResNet50 architectures, respectively.

Table 4.13: Results of VGG16-based CNN.

| VGG16 based Method | | | | | |
|---|---|---|---|---|---|
| Datasets | Acc | Cracked | Non-Cracked | No. Image (Training, Testing) | Batch Size |
| D4K | 97.7 | 97.7 | 97.7 | (360, 88) | 8 |
| DHD | 96.6 | 97.7 | 95.5 | (360, 88) | 8 |
| DG | 97.2 | 97.2 | 97.2 | (1434, 360) | 6 |
| ALL | 96.4 | 95.2 | 97.6 | (1072, 264) | 8 |
| Concrete (80/20) | 99.8 | 99.9 | 99.7 | (32000, 8000) | 8 |
| Concrete (60/40) | 99.7 | 99.7 | 99.8 | (24000.16000) | 8 |

Table 4.14: Results of ResNet50-based CNN.

| ResNet50 based Method | | | | | |
|---|---|---|---|---|---|
| Datasets | Acc | Cracked | Non-Cracked | Image Nr. (Training, Testing) | Batch Size |
| D4K | 99.1 | 98.2 | 100 | (360, 88) | 8 |
| DHD | 97.7 | 95.9 | 99.6 | (360, 88) | 8 |
| DG | 98.3 | 98.0 | 98.6 | (1434, 360) | 6 |
| ALL | 96.9 | 96.5 | 97.3 | (1072, 264) | 8 |
| Concrete (80/20) | 99.88 | 99.95 | 99.8 | (32000, 8000) | 8 |
| Concrete (60/40) | 99.7 | 99.5 | 99.9 | (24000.16000) | 8 |

These results unsurprisingly demonstrate that CNN models outperform all the hand-crafted feature models on all the datasets. In addition, the differences between specificity (Non-Cracked) and sensitivity (Cracked) are very small compared to the handcrafted methods, where a significantly higher specificity was observed across different methods. The high performance of the CNN can be explained by the fact that CNN can pick various features at different scales, which discriminates the cracks better than handcrafted methods based on single feature types. The differences between the performances of the two CNN models are marginal in each dataset. The results also show little difference between adopting different training/testing protocols used on the concrete dataset. However, the superior performance of the CNN models over the hand-crafted features must be considered in the context of their respective computational costs and requirements.

## 4.6 Discussions

In the following, we compare and discuss the classification performances of the different features. The experimental results presented in section 4.5 demonstrate the feasibility of building excellent, safe, and cost-effective software models based on ML algorithms for automatic crack inspection of glass façades and concrete structures, reaching a high level of performance. The single hand-crafted feature models use one type of image feature to classify the input image, while the CNN models use feature maps consisting of various textural and structural features at different scales. Among all models presented in this work, CNN models based on transfer learning performed best on all datasets. The pre-trained models learned different features, such as lines, corners, colours, and shapes, which may or may not be relevant to the shape (structure) of the cracks. In the case of the glass panels, CNN may use structural objects for the discrimination that appear on the glass surface due to the reflection of other nearby objects (e.g., trees, neighbour buildings etc.). Concrete surface cracks, on the other hand, are confined

in restricted regions of the image, which may lead to CNN detecting textures outside the crack region for discriminations unrelated to the actual cracks.

While the HOG model outperforms all other hand-crafted single-feature models on concrete with an accuracy of only 3% behind the CNN model, the ULBP model, among all other hand-crafted models, achieves the highest performance on all four glass datasets. This behaviour can be explained by the fact that glass crack features are best modelled by the local pixel intensity changes around the cracked regions used by LBP. In contrast, concrete crack features can be better extracted by directionalities (gradients in vertical and horizontal directions) of pixel intensity changes around the cracks used by HOG. The performance of the linearity model on concrete with 92% accuracy is not far behind, while the connected pixel configuration-based method is the least performing model, although it achieves accuracies as high as 88% on glass datasets. Another interesting point is that, in general, the achieved accuracies by all models are better the higher the image resolution. This observation is visible by comparing all model's performances on the D4K and DHD datasets which are images of the same glass panels at different resolutions.

The results also demonstrate that, for each method, partitioning the input image before feature extraction enhances performance to varying degrees of significance. Another observation is that methods based on analyzing the grayscale pixel intensities, such as ULBP and HOG, outperform other methods based on analyzing the edges extracted from the grayscale images like linearity, curvature or connected pixel configurations. However, making classification decisions based on the entire grayscale image runs the risk of being influenced by potential artefacts. Such artefacts include reflected objects or the appearance of objects behind glass panels, present on the glass surface, or by imperfections, such as dirt, paint, or uneven surfaces, present on concrete surfaces, which are unrelated to the actual cracks. In other words, other objects on the glass surface may contribute to the classification decisions rather than the actual texture distortions caused by the real crack segments. This means that if some details on the input image can be removed through, e.g., pre-processing operations while maintaining the details around the edge and line segments, then using texture features such as HOG and ULBP may result in better classifiers. A pilot experiment to test this idea is conducted in section 4.6.1.

Another interesting point, which is not being thoroughly investigated in this work, is the question of fusing different types of features. In the proposed solutions, only individual features are taken on for efficiency (perhaps the best-performing ones, i.e., ULBP and HOG). Nevertheless, fusion at the feature and decision levels can be promising in further improving the performance, given that the different individual features proposed in this work complement each other.

Further observation of the misclassification cases for the various methods shows that the two CNN models had only one joint glass and three joint concrete misclassification cases. On the contrary, ULBP and HOG models share many misclassified cases in glasses and concrete. There are minor same

misclassification cases between any of the two CNN models and the HOG or the ULBP schemes for the glass datasets and, to some extent, for the concrete dataset. In the case of the hand-crafted feature models, these observations can be explained by differences in the extracted texture features, whereas in the CNN models, it can be caused by the differences in the number and structure of convolution layers which may result in some differences in the learned structural features.

Finally, as explained earlier, the hand-crafted model training/testing experiments could be run on a cheaper computer using 5-fold cross-validations due to their lower computational costs. On the other hand, training and testing the CNN models using the much larger dataset of concrete surface images could only be run on the more expensive and powerful PC with 64GB RAM and two GPUs on board. Moreover, although a 1-fold 80%/20% (60%/40) training/testing protocol on the concrete dataset was used to test ResNet50, the task took 64 mins (48 mins). The corresponding execution time for VGG16 was 28.2 mins (20 mins). In contrast, the execution times for running the 5-fold cross-validation on all glass datasets were doable, although it took much longer on a much bigger concrete dataset. Accordingly, there would have to be a trade-off between accuracy and efficiency when an automatic crack inspection is considered for deployment onboard UAVs.

Finally, because the Hough transform can detect objects such as lines and circles of different shapes, we attempt to detect line-like objects from glass and concrete crack images using Hough transforms in the initial stages of this project. However, it did not lead to any reasonable results due to the complexity of the method.

### 4.6.1 Pilot Experiment on ULBP from Edge Segments

In the following, we describe using the ULBP feature extracted only from the crack edge segments. The ULBP (56) codes (excluding the 0 and 255 codes) are extracted from ED edge segment regions of the grayscale image and used to form a histogram for input to the classifier. For the sake of comparison, the results of the pilot experiment (ULBP + ED) are presented alongside other hand-crafted features in Table 4.15. Surprisingly, the results of the ULBP-based method applied only on the edge segments (ULBP+ED) are consistently worse than those applied to the whole image. Nevertheless, the results are still slightly better than the other two hand-crafted features of linearity and 3PixelConf.

Table 4.15: Comparison of the proposed features.

| Datasets | ULBP | ULBP + ED | 3PixelConf | Linearity |
|---|---|---|---|---|
| D4K | 91 | 88 | 85 | 81 |
| DHD | 91 | 77 | 74 | 76 |
| DG | 88 | 86 | 87 | 86 |
| ALL | 87 | 82 | 84 | 83 |

One possible reason for the drastic deterioration of the results using ULBP of the edges could be related to the ED edge detector. As explained before, the ED algorithm produces edges of one pixel wide,

meaning that we consider only ULBP codes laying on the edge segments and ignore all other surrounding pixels of the edge for the ULBP histogram calculations. Since the LBP algorithm is susceptible to intensity changes in the local neighbourhood, considering pixel intensities on thin crack edges reduces the discrimination power of the ULBP-based method. However, more research is needed to confirm this interpretation, e.g., by expanding the ED edges by some morphological operations or using other edge detectors such as Sobel and Canny, producing thicker edge segments including more edge neighbouring local pixels.

### 4.6.2 Pilot Experiment on Bayesian Classifier based on Connected Pixel Configurations

As a further pilot study, we attempted a Bayesian classifier based on connected pixel configuration for discrimination of the glass cracks. First, we use a Bayesian classifier to predict the label of the input image based on one connected pixel configuration at a time and then fuse the decisions made by 64 connected configurations. For realizing the Bayesian classifier, the mean of the frequency of the individual pixel configurations is calculated over all training cracked and non-cracked sets. In the testing step, the mean of the individual pixel configurations is calculated from the testing image and compared to the two-target class mean frequencies calculated previously from the training set. The predicted class of the particular pixel configuration is determined depending on how close its mean frequency is to the two training mean frequencies of the two target classes. Finally, the predictions of all or a subset of the 64-pixel configurations are fused to predict the final image label. As presented in Table 4.16, the prediction accuracies were significantly lower than those using kNN classifiers using the same 3PixelConf configurations.

Table 4.16: Comparison of the proposed features.

| Datasets | 3PixelConf (Bayesian Classifier) |
|---|---|
| D4K | 70 |
| DHD | 72 |
| DG | 70 |
| ALL | 71 |

## 4.7 Prototype for the Automatic Glass Façade Cracks Recognition

The following presents a prototype tool for automatic crack recognition in glass façades and concrete surfaces. Although the tool could be used to recognize concrete cracks, it could not be used due to the lack of a recorded video of concrete cracks. The tool is developed in C#.net as a side product of this research. The tool takes a video of a surface to be inspected and shows the individual video frame IDs recognized as cracked (in red) and non-cracked (in green) (see Figures 4.21 and 4.22). The prototype is based on the 3PixelConf method, but any of the proposed methods can be easily integrated. The inspected image (on the left of the main user interface) and its ED edges (on the right) are visualized in real-time. The menu button LoadVideo is used to load a pre-recorded video. The video includes 24 frames per second, depending on the video format. It is not necessary to process every 24 frames in a

second for crack recognition. Therefore, the menu Step can give the interval of the frame number to be processed, i.e., it specifies how many frames should be skipped each time a new frame is processed. The text box on the left show general information regarding the input video and the video format.

Each video frame takes roughly 0.5 seconds for crack inspection, depending on the feature type, PC, and image/video resolution. The prototype has the potential to be installed on board a UAV for real-time crack inspections with further improvements, especially in processing times.

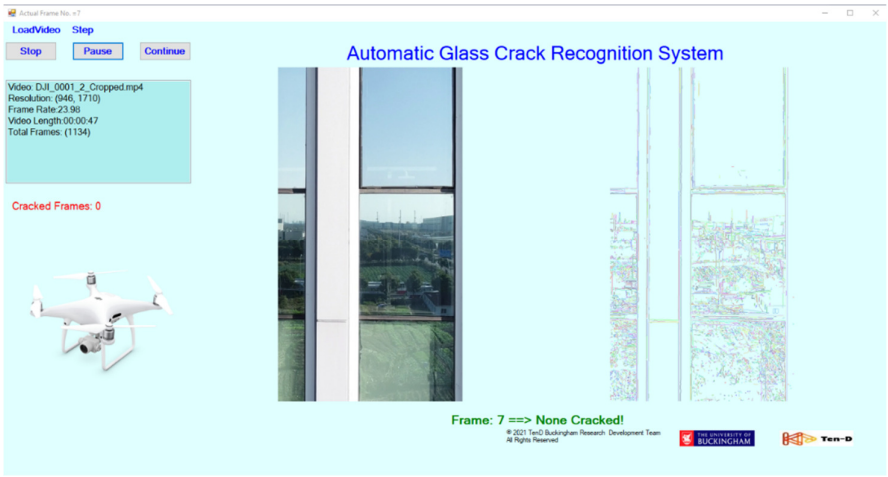



Figure 4.21: Prototype: Automatic glass façade crack recognition (no cracked glass panel recognized).


Figure 4.22: Prototype: Automatic glass façade crack recognition (cracked glass panel recognized).

## 4.8 Conclusion

This chapter presented novel solutions for automatic glass and concrete crack recognition. The solutions are based on several purposely extracted features using linearity, curvature, fixed-length connected pixel configurations, ULBP, and HOG to describe crack texture changes and shapes. The extracted features are simple, of low dimensionality, and lead to robust classifiers to be trained from the feature

spaces. Additionally, two CNN models based on transfer learning are introduced, and their test results are compared with the hand-crafted feature-based methods. DL-based methods achieved the highest performances; however, the simpler and less resource-demanding handcrafted feature-based methods were not far behind, making them suitable for installation on mobile devices. The performance of various handcrafted methods is improved by partitioning the input image. Evaluation results show that each feature type can distinguish cracked glasses or concrete from non-cracked ones, with LBP-based features giving the best accuracy on glasses and HOG on the concretes.

This research fills an apparent gap in the current work for glass façade inspections from outside the buildings and concrete surface inspections. The feature extraction and the classification are efficient, evident from the performance of the prototype software. However, the methods are evaluated on relatively small datasets collected from a small number of sources, as in the case of glass cracks. Moreover, the methods only predict the class of the entire glass panel or concrete plate without giving the severity of overall or individual cracks.

The next chapter discusses our second case study of abnormality recognition in medical images. More specifically, we present automatic border irregularity recognition methods for thyroid cancer lesions from ultrasound images whereby some of the features such as ULBP, HOG and HOL described here are reused, and various new ones are introduced.

# Chapter 5: Morphological Feature Analysis for Thyroid Nodule Border Irregularity

This chapter aims to develop automatic AI models for recognising thyroid nodule border irregularity from US scan images using morphological features from the nodule border. Automatic cancer diagnosis by analysis of radiological scan images of body organs/tissue is a growing area of AI research. For AI diagnostic schemes to be accepted and embraced by the clinical community, AI algorithms need to align their predictions with certain indicator parameters/signs of malignancy used by expert clinicians in their diagnostic efforts. One of these signs relevant to this thesis work is the nodule border abnormality in the geometrical concept of curve irregularity. In this and the next chapter, we investigate, develop and test the performance of traditional and deep learning AI algorithms for classifying thyroid nodule borders in terms of irregularity as an abnormality using the US scans of the nodule. As much as possible, we will avoid the heavy burden of accurate manual/automatic segmentation of the nodule border and attempt to rely on the geometric meaning of irregularity based on interpolating a less-than-accurate border curve using a sufficiently small set of border points marked by a radiologist. In this chapter, we shall focus on AI models of irregularity based only on the interpolated border from ROI points, while in the next chapter, we shall develop AI models of irregularity recognition by analysing texture variations around the interpolated lesion border. Figure 5.1 give an overview of the various methods presented in this chapter.

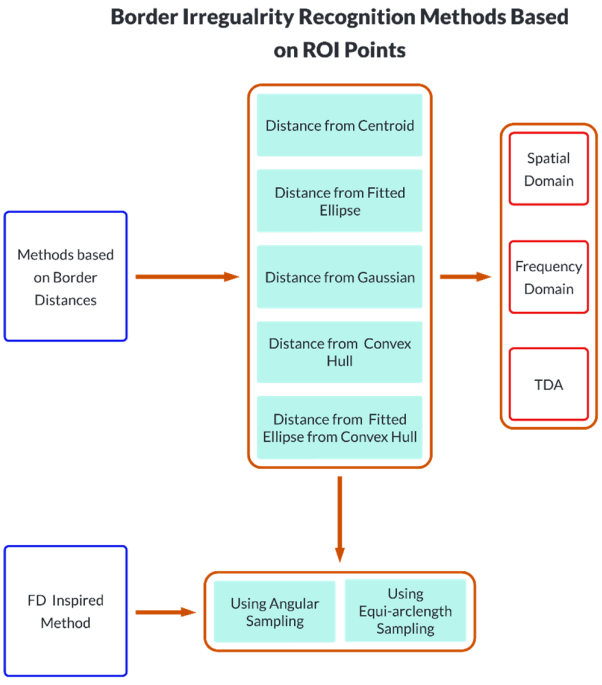



Figure 5.1: Summary of the Proposed Methods based on ROI points.

We first present a method for border interpolation from ROI points in this chapter. Then describe several methods based on border distances to different reference shapes. The distances are analysed in the spatial and frequency domain and by TDA for irregularity recognition. In addition, a method inspired by FD is proposed before the results are presented and analysed.

## 5.1 Introduction

Using US imaging in medical diagnostics is growing fast; however, there are various challenges in analyzing US images due to generally being of low contrast exacerbated by the presence of acoustic shadows and suffering from non-uniform distribution of speckle-noise. Nevertheless, deploying US tissue/body scan images for early detection/diagnostic purposes and screening programs is cost-effective and has obvious benefits. US imaging devices use high-frequency sound (ultrasonic wave) transducers to send sound waves to the body's inner organs and build an image from the reflected sound from the different body tissues. Due to the physical properties of the sound waves, unlike natural images, US images are complicated to interpret. Therefore, US images need careful examinations from experienced doctors and radiologists to determine the borders of different body organs and tissues. These difficulties led to the need for ML models to support doctors in diagnosing different cancer signs, including border irregularity of cancer lesions. For any ML methods to perform well in the assessment task of cancer lesion border irregularity, the exact borderlines of the lesion must be known.

Compared to benign cells, malignant cells have an accelerated cell cycle resulting in invasive growth and changes in cellular surface [147]. The onset of cancer results in gradual changes in the tissue texture that have different characteristics to the growth of benign tumours manifested as significant abnormalities in the corresponding US scan images. Such abnormalities are influenced by differences between the size/shape/action of cancer cells and non-cancerous cells, as well as the tissue/organ inter-cellular configuration, which facilitates intercellular control of tissue colour cell behaviour. Lesion border shape/smoothness/irregularity is among the list of cancer signs internationally adopted by various cancer-type imaging recording and data systems as indicators of malignancy. Although cancer lesion border irregularity is correlated to malignancy, neither every malignant tumour has an irregular border nor every benign tumour has a regular border [148].

Our reported handcrafted feature AI algorithms have been developed using datasets of US scan images that are collected from hospitals in Shanghai/China. The ROI in the US images is marked by a set of points to indicate the nodule border. The nodules are not segmented; consequently, any border irregularity method must rely on the marked ROI points. The number of ROI points is rather small and, in some cases, not precise enough to draw the exact border lines of the nodules. Instead of attempting to deploy manual or automatic border segmentation, we opt to estimate the nodule border naively by using bi-Cubic spline interpolation of the input ROI points. This interpolated border curve estimates the

border contour and will be used to compute shape descriptors of the morphological properties for nodule border irregularity discrimination.

Our proposed methods for border irregularity assessments are divided into two approaches: (1) methods using only ROI points to analyse the border and (2) methods analysing the texture features around the border. The methods based on the first approach are presented in this chapter, while the methods based on the second approach are presented in the next chapter. The next section introduces a literature review on cancer border irregularity signs and related works and describes the TI-RADS risk assessment system widely used for thyroid cancer assessment. The datasets used for the evaluation are described in section 5.3. Section 5.4 presents the proposed AI models for border irregularity recognition from interpolated borders. The experimental setup and results are presented in section 5.5. Section 5.6 presents the result's analysis and recommendations before concluding the chapter.

## 5.2 Background and Literature Review

In this section, we first introduce the TI-RADS score-based system widely used for thyroid diagnosis and then describe various works reported in the literature on various cancer border irregularities and related works.

### 5.2.1 TI-RADS Risk Assessment System

Thyroid Imaging Reporting and Data System (TI-RADS) is a risk-stratification system helping doctors to classify thyroid cancer into malignant and benign nodules by assigning scores to their different appearances in the US images [149]. Similar systems developed for other types of cancer, such as BI-RADS for assessing breast cancer US images [150] and ABCD rule of dermatology widely used system for skin mole assessment [151]. Thyroid nodules are frequent findings in neck US screening, but most of them are benign. Consequently, many nodules need to be sent to the lab for a biopsy examination to prove they are harmless. Therefore a non-invasive system such as TI-RADS is essential to provide a standardized guide to support practitioners and doctors for malignancy assessment of thyroid cancer. The degree of malignancy given by TI-RADS is categorized into benign, minimally suspicious, moderately suspicious, and highly suspicious based on different US features. The five feature categories: composition, echogenicity, shape, margin, and echogenic foci, are extracted from the US images and evaluated to calculate the final score. The margin sign (or feature) is the concern of this part of the thesis and deals with assessing the irregularity of the border of the nodule. The guide system assigns scores to the various features, with more suspicious features given higher scores. Figure 5.2 below presents the TI-RADS scoring system where the examiner needs to choose one of the features from each of the first four categories and all the features that apply from the last category. Summing up all the points gives the level of malignancy which starts with TR1 (Benign) and ends with TR5 (highly malignant). The practitioners can then recommend biopsy or US follow-up according to the TR level

of the nodule in addition to the nodule's maximum diameter. In the following we describe each of the features.

**The composition feature** describes the soft tissues and fluids content and their proportions in the nodule [152]. It is divided into several subcategories: cystic (entirely fluid-filled), almost completely cystic (more than 50% fluid), spongiform (filled with tiny cystic spaces), solid (consists of entirely soft tissues), and almost completely solid (consists of more than 50% soft tissues).

**Echogenicity** is the ability of the body tissues to reflect the US waves [153]. The US structures are characterized according to their echogenicity levels into categories; hyperechoic (white on the screen), hypoechoic (grey on the screen), and anechoic (black on the screen). A higher number of points are assigned to the grey (hypoechoic) regions, while black (anechoic) regions become no points according to TI-RADS (see Figure 5.2).

**The shape** represents the ratio of the two axes of the thyroid nodule; three points are given if the nodule is taller than wide; otherwise, no points are given.

**The margin feature** is concerned with the boundary of the nodule, which is categorized into smoothness, irregular, lobulated, ill-defined, halo, and extrathyroidal margins. The smoothness shows how well-defined or uninterrupted the edge of the nodule boundary is. An irregular margin which is the focus of this part of the work, shows whether the outer border of the nodule is jugged, spiculated, or has sharp angles with or without protrusions. The protrusions may differ in size and number. On the other hand, lobulations are rounded border protrusions variable in number and size. The nodule border is categorized as ill-defined if it is difficult to distinguish its border from the adjacent tissues. The halo margin represents a dark rim around the border of the nodule, partially or totally covering the margin. The extra-thyroidal extension is the last category in the margin feature, showing the nodule's extension to the neighbouring soft tissues and, therefore, highly correlated with the malignancy of the nodule.

**Echogenic foci** is the last feature category, which refers to regions of higher echogenicity. Punctate echogenic foci "Dot-like" foci become the highest score of three points. When calcification occupies the periphery of the nodule, it becomes peripheral calcification. The large calcifications inside the nodule forming macrocalcification indicate increased malignancy risk. A comet-tail artefact caused by reverberating acoustic waves has no associations with malignancy.

The margin feature is particularly important since it includes margin irregularity which is the focus of the current work. The irregular margin or lobulations in the nodule border indicate aggressive nodule growth highly correlated to nodule malignancy.

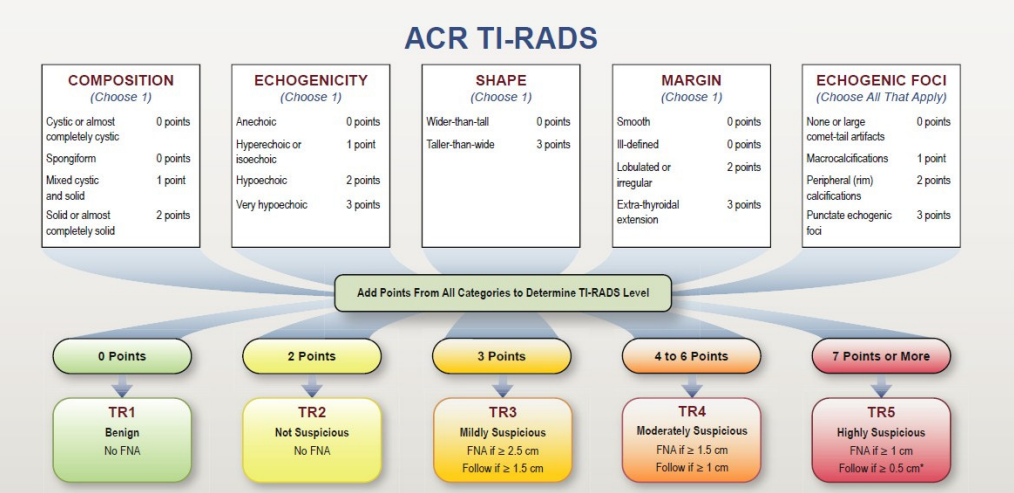



Figure 5.2: Sonographic features and associated points according to the American College of Radiology Thyroid Imaging Reporting and Data System, or TI-RADS [149].

### 5.2.2 Related Work on Border Irregularity

This section reviews the existing work on border irregularity of thyroid and other types of cancer in order to identify the main approaches. Only a small number of studies have been reported for the border irregularity of thyroid cancer, compared to numerous works on the assessment of irregularity in other types of cancer, such as skin moles and breast cancer. We first introduce the works on thyroid and breast cancer since they are both based on US images and then present the works on border irregularity diagnosis of skin mole from dermoscopic images.

#### 5.2.2.1 US Thyroid and Breast Cancer Recognition

In the following, we describe some of the works reported on thyroid and breast cancer recognition. We will focus more on reported works on breast cancer since much more research is being reported than on thyroid cancer. We remind the reader that most of the works reported are not concerned with classifying the cancer border into regular and irregular borders; instead, the classification is done into malignant and benign using different features, including border irregularity.

- **Thyroid Cancer Recognition**

The first paper reviewed in this part classifies US thyroid nodules into smooth (regular) and irregular borders by analysing nodule margin characteristics [154]. Different geometrical features (*Convexity, Solidity, Ratio aspect, Compactness, Circularity, Disperse, Tortuosity, and Rectangularity*) are extracted from the nodule margin and used for classification. Image noise and artefacts are removed using an adaptive median filter and Speckle-Reducing Bilateral Filtering (SRBF). The nodules are automatically segmented using morphological operations and active contour or snake. Methods performance on 144 images of 64 smooth and 80 irregular US images achieved an accuracy of 92.30%, sensitivity of 91.88%, and specificity of 92.73%. However, the source and description of the dataset

used in the experiments are not given. In another work, the box-counting FD method is used to analyse thyroid ultrasound images using Fractalyse Analysis 2.4 software [115]. A range of FD values is calculated for benign and malignant thyroid images. Different ranges of FD values were observed for benign and malignant thyroid cancers, which suggests the FD method is a good indicator for benign and malignant cancer prediction. The methods are tested on a publicly available dataset of thyroid nodules [155]. A fractal Brownian motion is used to measure the roughness of pixel intensities of thyroid nodules for classifications of benign and malignant nodules [156]. The US images were pre-processed by basic noise-reducing morphological operations and histogram equalization to standardize the grey levels in the US images before calculating the fractal texture features. Experiments on 60 US thyroid images show significant differences ($p < 0.05$) by comparing FD values of benign and malignant nodules using the Mann-Whitney U test, which shows the high efficacy of using FD for roughness estimation of thyroid nodules. The irregularity of the border region can reflect the surface roughness of three-dimensional images where pixel intensity represents the third dimension, making the use of box counting FD and Fractal Brownian Motion (FBM) potentially interesting for border irregularity assessment.

- **US Breast cancer Recognition**

Some works on border irregularity assessments for breast cancer are presented in the following. Irregularity features are extracted from the lesion boundary of breast cancer in US images and used for malignant and benign classification [157]. Fourier coefficients are calculated from the distances of contour points to the centroid of the lesion and used as an irregularity feature vector after transforming to a scale and start point invariant. The feature is combined with multiple other sonographic features. In total, 290 features are extracted from the ROI border and texture variations in the lesion and combined into one feature vector using the staked vector approach for the classification. The radiologists manually segment the lesion boundaries. The method was tested on 1599 benign and 2508 malignant cases, achieving accuracy, sensitivity, and specificity of 94.9%, 0.941%, and 0.958%, respectively. The features' normalisation and discriminability levels should have been considered since many of them are fused at the feature level. We have used a similar FFT transformation-based feature extraction from border distances; however, we discriminate between regular and irregular nodule borders in addition to using different features extracted from the Fourier spectrum (see section 5.4.9.1). Another similar work has been done for malignant and benign recognition of breast lesions from US images using Fourier Transform [158]. Fourier Irregularity Index (FII) is calculated from Fourier coefficients using distance-based shape descriptors. The radial distances are measured from each contour point to the image origin. The calculation of FII is based on the observation that the energy distribution in the high-frequency regions of Fourier coefficients is greater in cases of irregular than regular lesion contours. The results of the proposed method are compared with previously reported methods based on compactness, FD, Fourier factor, and the fractional concavity extracted from the

contour of the lesions. The FII method outperformed all four methods achieving an accuracy of 96%. Both above-described FFT-based methods rely on manual segmentation of the lesion boundary in contrast to our work based on border estimation from ROI points.

Pereira et al. [159] extracted seven features from the contours of breast cancer lesions and ranked them according to their high malignancy discriminability using Stepwise Linear Discriminant Analysis (SLDA) and Mutual Information (MI). After feature extraction from the convex polygon and the normalised radial distances of the lesion contour, MI and pairwise feature correlations were computed. The experiments were done on a dataset of 77 malignant, and 69 benign tumours with the borderlines segmented semi-automatically. Even though the features were not used directly for classification (by demonstrating their accuracy, for example), their relevance to the malignancy is described. In another work, several features are derived by measuring the radial length of the tumour contour for benign and malignant breast cancer classification [160]. Stepwise logistic regression (SLR) is used to assess the features' significant (p-value) correlations in discriminating 40 benign and 71 malignant US images segmented manually. The three most significant features are selected and used for the classification, achieving accuracy, sensitivity, and specificity of 91%, 97.2%, and 80%. The proposed method relies on manually sketched nodule contours, which might not be accurate enough in addition to the relatively small dataset. The work in [161] compares linear and nonlinear machine learning algorithms for classifying malignant and benign thyroid nodules. Several features, including margin and shape (regular and irregular), are extracted from US images and ranked according to their discriminability. The experimental results show no significant performance differences between the two types of machine learning algorithms. The performance of machine learning algorithms is compared using the mean of the area under the curve (AUC), and the confidence interval (CI) of 1000 iterations of training/testing splits of the dataset. The nodule shape was ranked second important feature after the calcification according to their mean decrease in Gini. This shows the relevance of the border shape to the malignancy of tumours.

#### 5.2.2.2 Skin Mole Cancer Recognition from Dermoscopic Images

The works on skin moles cancer recognition are discussed in this section, with some of them being presented in more detail due to their strong relation to our research on recognizing border irregularity of thyroid cancer nodules. Unlike the two above-described cancer types, skin mole assessments are based on the dermoscopic imaging modality, which is coloured photos of the skin mole, like regular photos. Nevertheless, the segmented border coordinates of the mole can be assessed for irregularity using the same machine-learning technics used in other types of cancer. Similar to TI-RADS and BI-RADS cancer diagnosis systems, the ABCD rule for dermoscopy is used as a guideline system for skin mole assessments.

The first work reviewed here is a new method proposed to detect skin mole border irregularity [162]. The method steps include image enhancement, mole segmentation, borderline function estimation, and irregularity assessment. Due to the strong synergy of the work to our distance-based methods for thyroid border irregularity recognition, the steps are explained in detail. The mole is rotated before building the borderline function by rotating the major axis of the mole at a certain angle α so that it is parallel to the horizontal line (see Figure 5.3 a). For building the borderline function, the bounding box around the mole and the centre of mass (the intersection points of the two diagonal lines) are drawn (see Figure 5.3 b). Then, the border distances are measured for the four border regions separated by diagonal lines. The distance function (see Figure 5.3 d) is derived from Euclidian distances measured from border points to the individual bounding box edges (red arrows Figure 5.3 c). Such distance functions are descriptors of the tumour morphology.

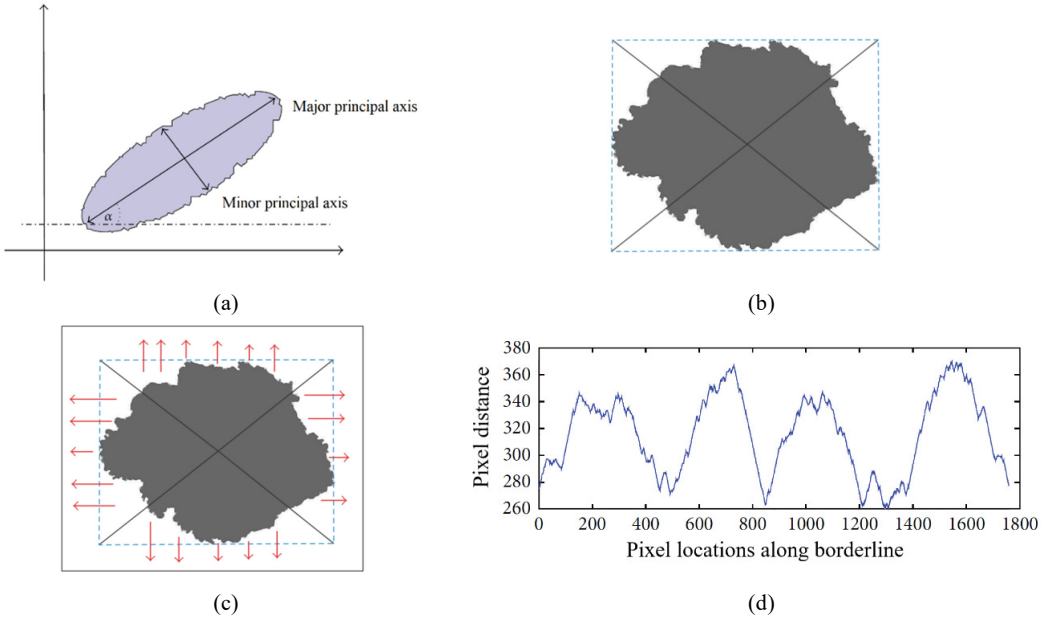



Figure 5.3: Borderline function for a skin mole (a) defining the major and minor axis of the mole (b) dividing the mole into four regions (c) border distance to the image edges (d) building distance function [162].

The final irregularity index is calculated by counting the turning points of the borderline function. A dataset of 350 skin mole images containing 280 benign and 70 malignant cases is used for the evaluations achieving accuracy, sensitivity, and precision of 92%, 91%, and 89%. The proposed irregularity measure showed superior performances over the state-of-the-art methods based on radial distances. Contrary to our work, the mole boundary is segmented automatically using a seeded region-growing algorithm. We use similar borderline distances to assess mole border irregularity; however, we measure the distances to different reference shapes (fitted ellipse, Gaussian shape, convex hull, and fitted ellipse from convex hull) in addition to the centroid as a reference point.

Six different FD methods are used and compared for irregularity assessments of skin moles from coloured photos (dermoscopic imaging modality) [163]. It is interesting to mention that the reduction of the effect of inaccurate mole border segmentation is attempted by assessing the pixel intensities around the borderline using the fractal Brownian function, which could present an interesting extension of our FD-inspired method presented in section 5.4.10. The FD based only on the borderline is compared with the FD based on the pixel intensities around the mole boundary. The evaluation results of the proposed methods on a dataset of seven benign and seven malignant mole images show differences in FD p-values calculated using different methods for normal and abnormal moles suggesting that FD is a good indicator of border irregularity. The methods proposed in [114] utilize the pixel intensities in skin moles to calculate FD. Different methods of Differential Box Counting (DBC), horizontally smoothed DBC (HDBC), vertically smoothed DBC (VDBC), and multi-fractal DBC (MULTI) are used to calculate FD, where MULTI-based FD performed the best.

Next, we describe the work presented by Lee [164] for border irregularity assessments of skin moles that has a great deal of synergy/similarity with our work in border irregularity assessment of thyroid cancer. The work categorizes the border along the mole boundary into textural and structural irregularities. The textural irregularity is small texture variations along the border, while the structural (also called global) irregularities are indentations and protrusions of the mole boundary. Structural irregularities suggest uncontrolled growth of the mole and, therefore, are highly correlated to the malignancy of the mole. The roughness of the mole boundary is measured using a simple index called Sigma-Ratio (SR) derived from a sigma value needed for a Gaussian filter to remove all concavities from the borderline divided by the length of the border. The protrusions and indentations are detected using zero crossings of the curvature values along the boundary. In the second irregularity index: extended curvature scale-space, the zero crossings of partial derivatives of the curvature along the border are used to detect all concavities along the borderline. The most significant (MSII) and overall irregularity (OII) indexes are calculated as new measures for all structural irregularities along the mole boundary. MSII represents the border's largest indentation/protrusion segment, while OII is the sum of all indices calculated from all detected indentations and protrusions. The two irregularity indexes with three other common shape descriptors, compactness index (CI), FD, and Structure Fractal Dimension (SFD), are evaluated on 40 automatically segmented moles. OII achieved the highest accuracy among all indices using the clinical evaluations of 20 radiologists as a benchmark.

#### 5.2.2.3 Ground Truth Predictions and Inter and Intra-observers Variability

Medical image analysis, particularly US cancer image analysis, does not have a unique ground truth, in contrast to many other applications. The work in [165] has experimented with inter and intra-observer variabilities in the description of US features of thyroid cancer, including nodule margin. In the study, benign and malignant nodules were analysed by five radiologists at six-week intervals. The Thyroid

Study Group of the Korean Society of Neuroradiology and Head and Neck Radiology (TSGKSNRHNR) guideline is used in the analysis instead of the TI-RADS scoring system. The study calculated the interobserver agreement between different radiologists using Cohen's kappa statistics (k) for different cancer signs. The degree of agreement for irregular shapes was low, giving k= 0.26, whereas the agreement for other cancer signs was generally higher. Another work describes the values and limitations of various sonographic features from thyroid US images [148]. The sensitivity of the different US features for detecting thyroid cancer malignancy is reported. A high sensitivity variation between 7% to 97% is reported for an irregular margin or boundary of the nodule. This suggests that even though border irregularity and nodule malignancy are strongly correlated, not all irregular border nodules are cancerous, and vice versa.

In our abnormality investigations, we observed the effect of this inter- and intra-observer variability on the performance of our schemes. Therefore, we shall develop our schemes by training samples labelled by one radiologist but will be tested with class labels annotated by other radiologists. This may help explain different views of the radiologists and consequently results in ground truth-dependent performances for our schemes. Note that inter- and intra-observers ground truth labelling is not a problem for our case study of building material cracks recognition.

#### 5.2.2.4 Summary of Literature Review

In general, the existing techniques for irregularity classification/assessment described above are based on measuring radial distances of the border points from a reference point or reference shape or edges such as bounding box edge and subsequent irregularity extraction from the distances function. Other methods use single feature values such as FD or CI from the cancer mass boundary to recognize irregular cases or are used for discriminating malignancy of cancer. The FD can be extracted from the lesion borderline or the pixel intensities around the border. Some methods use curvature with Gaussian smoothing of the borderline to detect protrusions and indentations for deriving an irregularity index. The reviewed border irregularity recognition has a welcome guiding influence on our investigations, but all assume the input of highly accurate manual/automatic lesion border segmentation. Thus, two approaches can be identified in the literature: (1) distance-based functions between the segmented borders to reference shapes/objects (lesion-box edges or lesion centroid) and (2) FD estimation of the lesion border. The distance function is further analysed in the frequency domain.

Our AI-based border irregularity investigations start, in the next section, with the process of constructing a naïve approximation of the thyroid nodule border using the input ROI points instead of an accurately segmented border. We shall discuss and illustrate the various shortcomings of this decision and ways of preparing this assumed nodule border to be used with the least amount of bias due to variation in the nodule size and number of labelled ROI points. We shall then consider various ways to define distance functions in terms of the interpolated nodule border, develop related methods for

irregularity recognition in both spatial and frequency domains, and conduct experiments to test their performances. We shall then develop FD-inspired approaches and discuss their performances with those of the various distance function-based approaches. Finally, we shall present a comparison of the performance of all the proposed methods.

## 5.3 Dataset and Data Preparation

Over the previous three years, TenD project participating hospitals generated two datasets of thyroid US scan images labelled with regularity/irregularity characteristics, whereby each nodule border is marked with ROI points, and no border segmentation was provided. Two datasets of 395 and 100 cases are used to evaluate the proposed methods. The first dataset DS(395), is used as internal training and testing set, while the second dataset DS(100), is used as an external testing set. The datasets are described in the followings.

### 5.3.1 Internal Dataset

The dataset was collected from one of the hospitals in China, and one specialized doctor marked the ROI points. The dataset was collected from 2020 to 2021 in Renji hospital in China (Shanghai) and labelled by two doctors, where the labels: Regular and Irregular were given.

Table 5.1: Internal dataset.

| **DS(395) Dataset** | |
|---|---|
| Label | No. of Cases |
| Regular | 181 |
| Irregular | 214 |
| Total | 395 |

The dataset is almost balanced, with 33 more irregular than regular cases, as seen in Table 5.1.

**Statistics of Internal Dataset:**

Variations in lesion size may have an adverse impact on the performance of the proposed ML methods. Therefore, in the following Figure 5.4, we show the distribution of the lesion sizes across the internal dataset DS(395). The sizes are calculated in pixels by simply counting all pixels lying inside the boundary of the lesion. The size range varies from 538 pixels for the smallest lesion to 185340 pixels for the largest lesion.

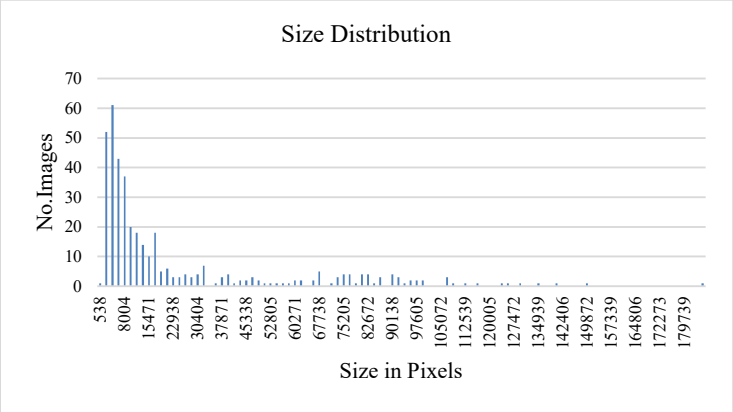



Figure 5.4: Size distribution of the lesions across the internal dataset.

As mentioned earlier, the number of the ROI points of the different cases in our dataset varies. Since the majority of our algorithms for detecting irregularities rely on the ROI points, it is interesting to know how the number of ROI points is distributed among our datasets, as shown in Figure 5.5 below.

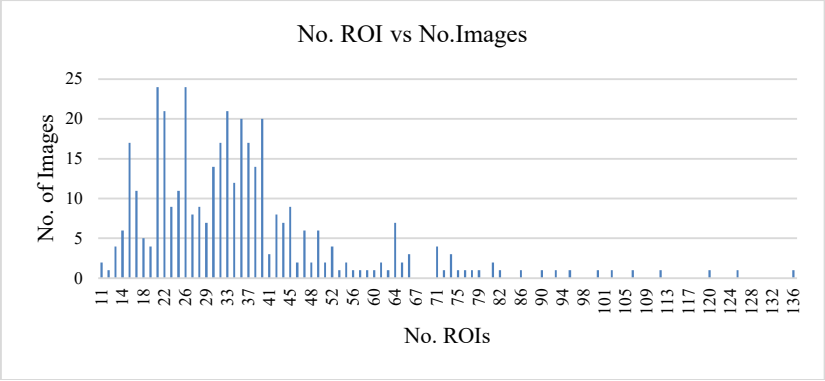



Figure 5.5: Distribution of the number of ROI points across the DS(395) dataset.

It is also interesting to know the distribution of the lesion sizes with respect to the two regular and irregular classes. Figure 5.6 below shows each class's size distribution in different colours, where the dots represent individual US cases. It shows almost equally distributed nodule sizes among the two classes, i.e., there is no correlation between the lesion size and the irregularity class.

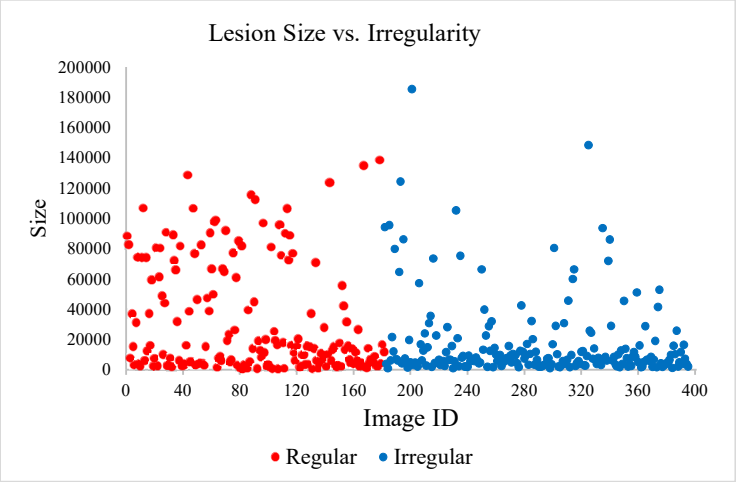



Figure 5.6: Size distribution among the two irregularity classes.

### 5.3.2 External Dataset

The dataset of 100 US thyroid cancer images is also collected from one of the hospitals in China and denoted as DS(100). Unlike DS(395), this dataset is provided with separate class labels assigned by three doctors (A, B, and C). Table 5.2 below shows the number of cases labelled as regular and irregular by each of the three doctors. Due to the different nature of the dataset and its small size, we use this dataset as external testing set to evaluate methods trained using the internal dataset.

Table 5.2: Three doctors' labels for DS(100).

| **DS(100) No. of Cases Regular and Irregular** | | | |
|---|---|---|---|
| Label | Dr A | Dr B | Dr C |
| Regular | 44 | 40 | 49 |
| Irregular | 56 | 60 | 51 |

**Statistics of External Dataset:**

Since three doctors label the external dataset DS(100), knowing how many labels are agreed upon between all and each pair of them is interesting. Generally, the number of agreed irregular cases is higher than the agreed regular cases (see Table 5.3), possibly due to a slightly higher number of irregular cases labelled by each doctor. Several graphic representations of the statistics of the external dataset similar to the internal dataset are attached in appendix A. The statistics show similar distributions of the cases regarding their sizes, the number of ROI points, and their class labels compared to the internal dataset.

Table 5.3: Agreed ground truths between the doctors.

| Agreed labels among all three doctors | |
|---|---|
| Label | No. of cases |
| Regular | 30 |
| Irregular | 43 |

| Agreed labels among doctors A & B | |
|---|---|
| Label | No. of cases |
| Regular | 30 |
| Irregular | 46 |

| Agreed labels among doctors A & C | |
|---|---|
| Label | No. of cases |
| Regular | 39 |
| Irregular | 46 |

| Agreed labels among doctors B & C | |
|---|---|
| Label | No. of cases |
| Regular | 37 |
| Irregular | 48 |

## 5.4 Proposed Methods

Figure 5.7 below shows an overall workflow of our methods for border irregularity recognition of thyroid nodules. As the first step, the borderline is approximated from the ROI points, and new fixed-length border points are sampled. Then, the methods either (1) use a distance function measured from the sampled points to different references for analyses in spatial and frequency domains before feeding into a classifier or (2) based on measuring the perimeter of the interpolated border at different scales (FD-inspired) an irregularity index is extracted for use as discrimination feature.

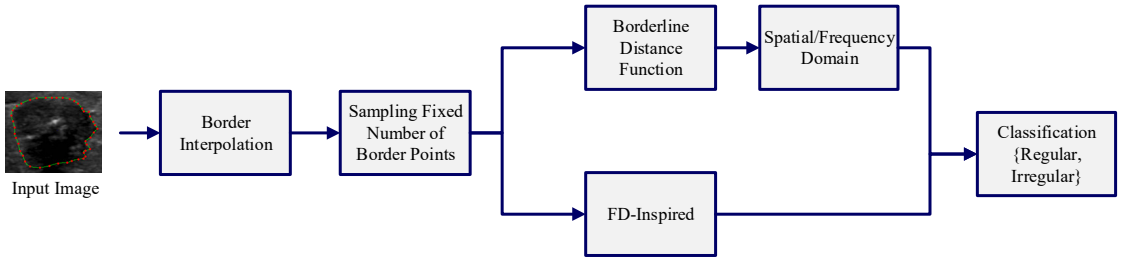



Figure 5.7: Workflow overview of the methods for thyroid nodule border irregularity.

In the following sections, we present the method for borderline approximation from ROI points with two sampling methods and then introduce our proposed methods.

### 5.4.1 Lesion Border Approximation

The use of AI for analysing US scan images of cancer in terms of disease-related abnormalities has only become possible in recent years. This fact explains the lack of sufficiently large datasets that have been prepared for machine learning. Although hospitals have been using US scan images for a much longer time, and some may have plenty of such images, the recorded images are not readily usable for training AI algorithms. If they exist, preparing such datasets requires manual, laborious, time-consuming, and error-prone lesion segmentation and labelling, while most health services have an acute shortage of well-trained radiologists and clinicians. Automatic segmentation for such existing datasets is unrealistic, as we still need sufficiently large samples to have been accurately segmented and labelled for the intended abnormality analysis. Hence, the TenD collaborative initiative opted to construct new

purpose-built datasets of US lesion scan images. Clinicians in various participating hospitals support preparing the scanned cancer nodules by clicking on a set of ROI border points and labelling cancer's various malignancy signs. For the speed of operation and due to the earlier mentioned challenges in segmenting the US scanned legions, we decided not to wait until a fully automatic lesion segmentation becomes available to conduct our research into border irregularity AI schemes. Instead, we use the clinician's labelled ROI to estimate an approximate lesion border curve that goes through the border using mathematically available interpolations techniques.

The most straightforward technique is linear interpolation, equivalent to the polygonal shape obtained by drawing straight lines between successive ROI points when scanned in a fixed orientation starting from the agreed starting point. However, it is highly unlikely that this result in a reasonable border approximation. Quadratic and higher-degree polynomial interpolations are possible and may better approximate the lesion border than the linear case. However, the higher the degree, the more computationally inefficient the interpolation becomes. Due to variations in the number of ROI points, the use of the well-known BiCubic spline interpolation is a more realistic compromise.

Cubic spline interpolation is a mathematical method used to smoothly connect (interpolate) a set of points, i.e., to create new points from the spline function of a set of existing points. It associates a set of cubic curves fitted to successive pairs of points with the condition of continuity and smoothness at the common points. The spline function is defined piecewise by cubic polynomials, provided in Matlab and Python libraries, and used in our border approximation methods. We need to remember that in relation to our use of cubic spline, we are not using any pixel intensity values, i.e., it is designed for points in a 2-dimension form. However, there are versions of cubic spline in 3 dimension space. Nevertheless, for simplicity and to avoid reliance on intensity values that could be variant on the ROI marked points, we use a 2-dimensional version of the cubic spline.

Several challenges must be addressed before training and testing any particular AI model on the input of the interpolated borders for determining lesion regularity/irregularity characteristics. One issue with using distances as feature vectors for input to an AI classifier is the different lengths of the distances function due to the different numbers of the ROI points for the different lesions. Since the classifiers require the input feature vectors of equal length, using the distance function approach necessitates re-sampling interpolated borders with a fixed number of points. We shall now describe two sampling approaches that output a fixed number of interpolated lesion border points.

#### 5.4.1.1 Equi Angular Displacement Distances-based Border Sampling

We describe our first interpolated border sampling method of equal angular displacement (Equi-angdisp). This sampling method first draws a circle; its centre is at the centroid of the ROI points, and its radius **r** slightly longer than the longest distance between the centroid and the ROI points (see white

dots in Figure 5.8 c). Then the circle is divided into n equal-sized sectors and determines the n points (red squares Figure 5.8 c, d, and e) on the interpolated nodule border that intersects the straight lines. For example, if n=90, the angular sectors subtend angles of 4° (360°/90=4°). Figure 5.8 illustrates this process for a nodule border of high irregularity characteristics where the individual figures show: (a) the original ROI points (red squares), (b) interpolated border from ROI points (green curve), (c-e) angular border point selection starting from 0° (red line) and runs in the clockwise direction (blue line), and (f) the final selected 90 points on the interpolated border. Our sampled points always start at the same location and run in the same direction since the first sector (red line) starts at the same angle of 0° and runs in the clockwise direction (blue line), which solves the issue of different starting locations and directions of the doctor's marked ROI points across our datasets.

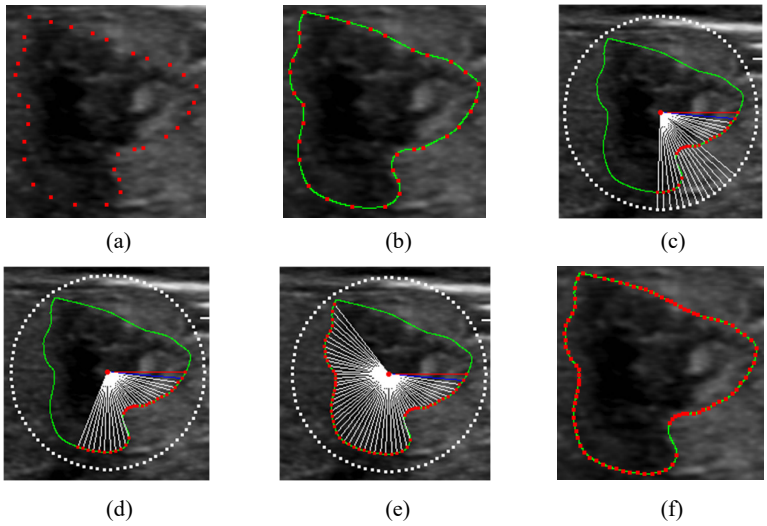
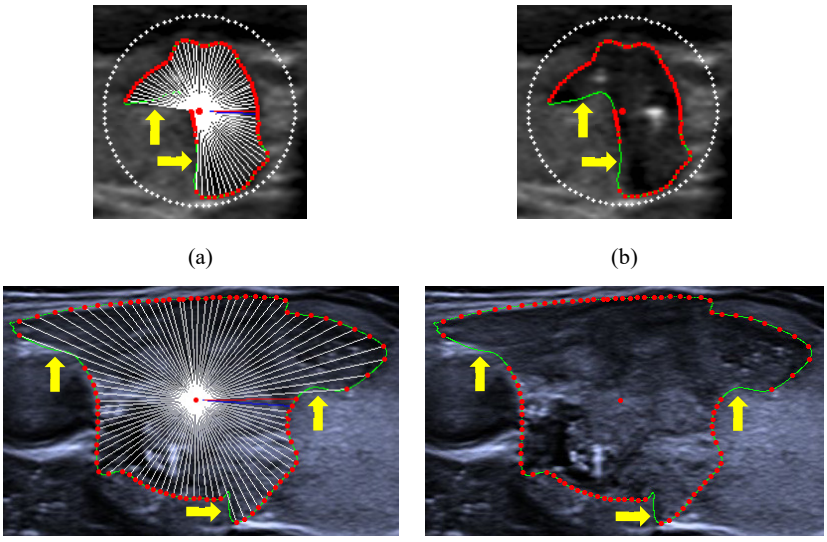



Figure 5.8: Sampling Interpolated Border Points using Equi-angdisp method.

This method works well for most images having elliptical shapes. However, for extreme shape lesions, it might miss some areas exhibiting irregularity characteristics or pick more points in one section of the border than others. Figure 5.9 below shows 2 cases of seemingly high irregularity characteristics. The yellow arrows show the significantly large border sectors of empty sample points.



(a) (b)

(c) (d)

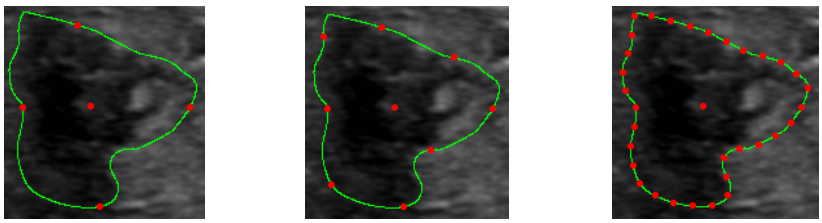
Figure 5.9: Some issues with the Equi-angdisp method for sampling interpolated border points.

#### 5.4.1.2 Sampling Interpolated Border at Equi Arc Length Steps

In the following, we describe our equal arc length (Equi-arclength) based method for sampling the interpolated border points. Simple Euclidian distances along the borderline are used to solve the abovementioned issues. As the first attempt to implement the method, the whole perimeter of the borderline is divided by the number of the required interpolated border points to get the distance between every two successive points (called step). Then, starting with the first point on the borderline, the interpolated border points are picked in multiple step distances. However, there is a small tolerance issue with the number of sampled interpolated boundary points depending on the size, shape, and required number of points. Drifts in the distance measurements between every two successive points due to the discrete nature of the digital images cause the error, accumulating along the borderline to a total error and increasing the longer the borderline or the higher the number of required points. Therefore, this way of sampling the interpolated border does not satisfy the requirement of an equal number of border points across the dataset for building the distance function.

A recursive bisection method of border point selection is used to overcome this issue. As before, we first measure the perimeter of the interpolated borderline. Then the middle point is selected by dividing the whole perimeter by 2. This procedure creates three points; the borderline starting and end points and the selected middle point. Now, the same procedure is recursively repeated on the two halves of the borderline; the middle and starting point on the one hand and the middle and end point on the other hand, creating for each of them a new middle point (see Figure 5.10). This procedure is repeated until the required number of points on the borderline is selected. Due to the nature of the bisection selection, the number of required points always needs to be $2^n$ where n is the number of the sections (i.e., if the number of sections n=6, the number of selected points = $2^6$ = 64 points). Figure 5.10 below shows a

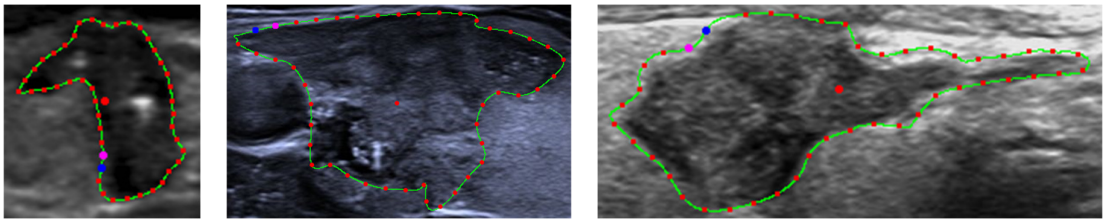
thyroid nodule with a different number of selected points at Equi-arclength using the bisection sampling method.



(a) 2 sections = 4 points (b) 3 sections = 8 points (c) 5 sections = 32 points

Figure 5.10: Bisection sampling procedure.

The bisection method for border point selections is precise in the number of selected points, opposite to the previous naive way of measuring the distance between two successive points. However, through the bisection point selection, the order of the selected new points along the borderline is distorted, which are rearranged to the correct order by assigning some indexes to the sampled points. The method still has issues with different starting locations and directions of border points, as seen in Figure 5.11 below. In the figure, the blue dot marks the starting location, while the purple dot shows the direction of the border points. In Figures a and b, the border point's direction is clockwise, while in Figure c is anti-clockwise. These differences are caused by doctors who originally marked the ROI points.

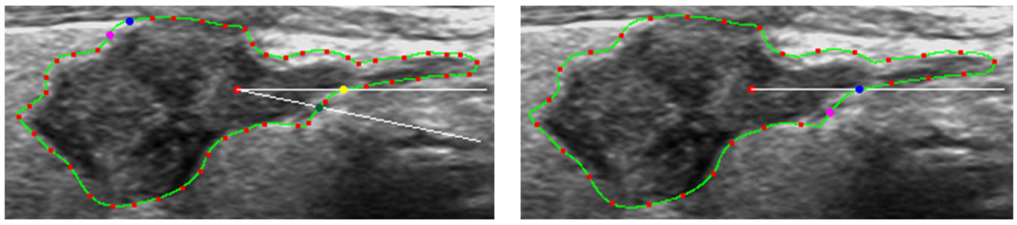


(a) (b) (c)

Figure 5.11: Sampling interpolated border points using Equi-arclength steps. Border points starting location (blue circle) to the direction of the purple circle.

A simple solution is to use angular distances to mark the border's starting point and then rearrange the whole border points from the new starting point. We first draw a line from the centroid outwards at a 0° angle (white line in Figure 5.12 a), crossing the lesions interpolated borderline. Then, the intersection point is marked as a new starting point (yellow point in Figure a). The points on the interpolated borderline are now rearranged from the new starting point. The next issue to solve is to correct the sampled border points sequence into one standard direction, which is clockwise in our case. Here, we draw another line in addition to the line for marking the starting point at, for instance, 25° and mark the intersection point on the borderline (dark green circle in Figure a). If the index of the green point in the sampled points sequence is bigger than the starting point (yellow point), the border points are in the right direction. Otherwise, the border points are in the opposite direction and need to be rearranged.

Figure b presents the lesion with a corrected border starting point and direction, where the blue dot marks the new starting point and the purple dot the direction.



(a) (b)

Figure 5.12: Correcting the border starting point and direction after sampling interpolated border using the Equi-arclength method.

### 5.4.2 Distance Functions for Border Irregularity Recognition

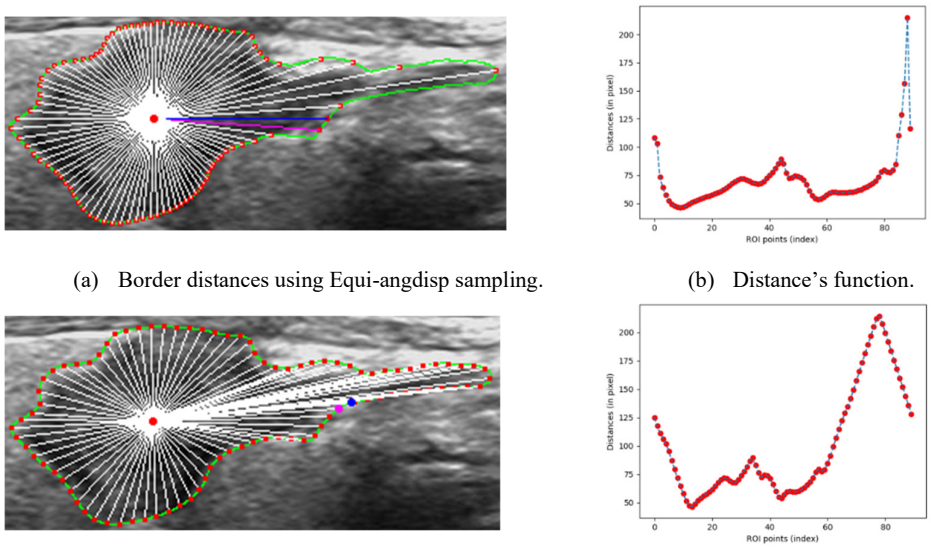
The concept of closed curve irregularity appears in diverse areas of science and art that rely on shape analysis, ranging from curve drawing and curve fitting to fractal shapes. Very few planar closed curves/polygons classes are perfectly regular, namely circles and convex polygons with equal length sides and fixed internal angles of $< \pi$. Cancer (benign or malignant) growth is highly unlikely to have such strictly defined regular shapes, and therefore we need to extend the list of regular shapes by including less restrictive conditions. Taking this relaxation of regularity conditions into account, we determine the irregularity characteristics of a planar closed curve by investigating the frequency of encountered changes to geometric parameters (e.g. direction, curvature, and discontinuity) computed when the curve is traced in a specific orientation. To some extent, such characteristics of curve irregularity can be measured by how much the curve deviates from bordering known "regular geometric" shapes such as triangles, rectangles, regular convex polygons, circles, and ellipses. Such regular shapes can be used as references to determine the irregularity characteristics of lesion borders.

A nodule-size-dependent set of ROI points determines our interpolated lesion border, and we use several ways of identifying regular reference shapes to define distance functions. As a single point, the centroid of the ROI points is the converging point of a sequence of concentric circles of decreasing radii and is a simple candidate for regular reference shape. Other ROI-dependent reference regular shapes that we will use in this chapter include: (1) the fitted ellipse of all the ROI points, (2) the fitted ellipse of the convex hull corners of the ROI points, (3) and Gaussian curves fitting of the ROI points. For the sake of completeness, we shall also include the ROI convex hull as a reference shape instead of using the lesion inclosing box edges used in the literature (see section 5.2.2.2).

In the remaining part of this section, we describe the method of defining these various distance functions, highlight the effects/shortcomings of using the corresponding reference shape, and discuss different ways of extracting fixed-size feature vectors to enable the use of AI for irregularity detection.

### 5.4.3 Distance Function from Centroid of ROI Points

The centroid can be calculated as a centre of mass or calculated by averaging all border point's x and y coordinates. The latter one is used in our experiments. First, the required number of points from the interpolated borderline are selected using any previously described border point sampling methods. Then Euclidean distances between the border points and the centroid are measured and used to build a distance function. Figure 5.13 below shows the distance function for the same nodule cases using both methods of Equi-angdisp and Equi-arclength for sampling methods. Their distance function graphs are slightly different but have similar turning points (see Figures b & d). The border points on the borderline are more evenly distributed using the Equi-arclength than the Equi-angdisp sampling method. The effect of the sampling method is more obvious around the elongation of the lesion on the right side. In the case of Equi-angdisp sampling (Figures a), the elongation is captured by one or two sampling points, while it is captured by more evenly distributed samples in the case of Equi-arclength sampling (Figures c).




(a) Border distances using Equi-angdisp sampling.

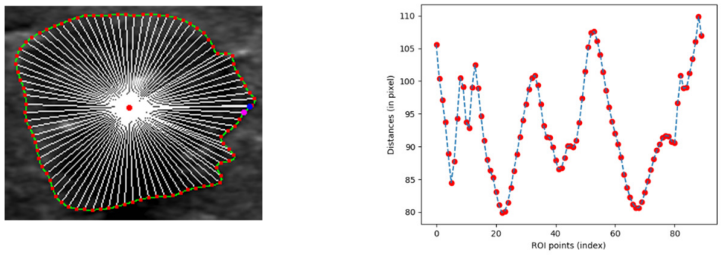
(b) Distance's function.

(c) Border interpolation using Equi-arclength sampling.

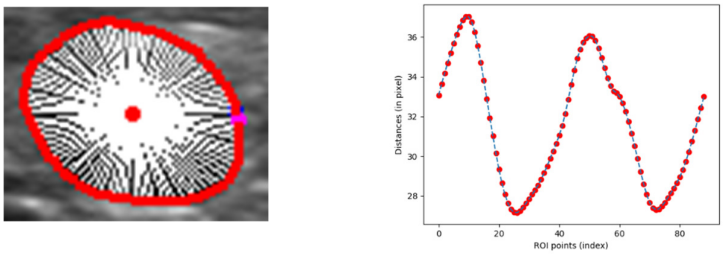
(d) Distance's function.

Figure 5.13: Building distances function from 90 interpolated border points.

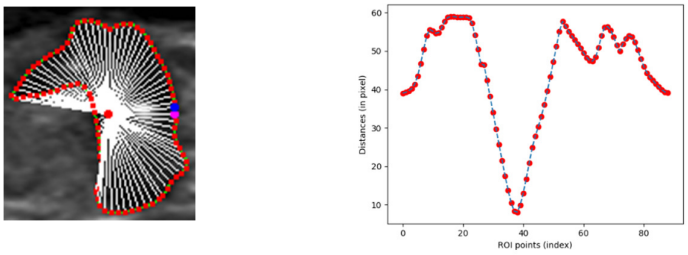
Figure 5.14 shows several regular and irregular cases and their distance function graphs using the more precise sampling method of Equi-arclength. We can observe fewer turning points in the distance graph of the regular case (Figures c & d) than in those of the remaining three irregular cases (Figures a, e, and g). Further, the turning points or minima and maxima of the distance function curve of the irregular cases are more frequent than the ones of the regular case, confirming abrupt turnings in the borderlines of irregular cases.



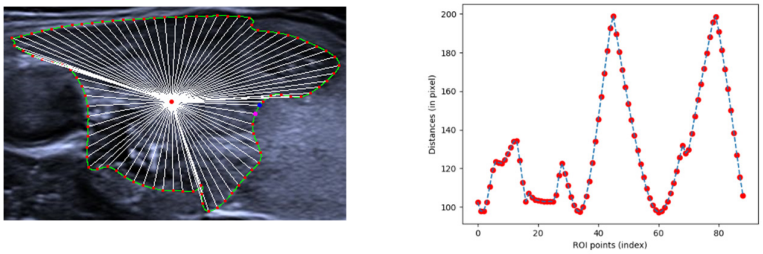

(a) Irregular case (b) Distance function




(c) Regular case. (d) Distance function.




(e) Irregular case. (f) Distance function.



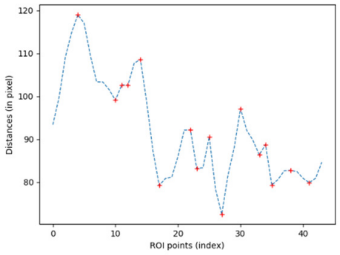

(g) Irregular case. (h) Distance function.

Figure 5.14: Distances function using centroid as reference for regular and irregular cases.

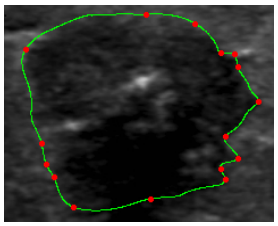
### 5.4.4 Distances Function for Irregularity Visualization

Another application of the distances function is its use for border irregularity visualization/localization. The visualization is done by examining the profile of the minima and maxima of the distance curve. For this, we will show some examples in Figure 5.15 that help design an appropriate analysis of the distance functions. Figure a shows the distance function of an irregular case using the original ROI points, i.e., without border interpolation, where the distance curve's minima and maxima (red pluses) are used to illustrate the irregularity locations along the borderline, as it can be seen in Figure b. In comparison, Figure c shows the distance function of the same irregular case calculated from 90

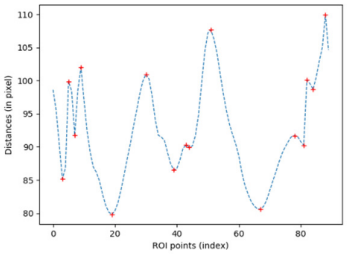
interpolated points on the border using Equi-angdisp sampling with irregularity locations marked by red strips, as in Figure d. Figures e and f show a regular case with marked irregular locations. From the three illustrated cases, we can observe (1) that the irregularity visualization using interpolated borderline is more precise than using the original ROI points and (2) that a higher number of irregularity locations are present in an irregular case than in a regular case.



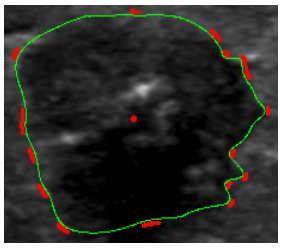

(a) Distance function of an irregular case with 44 original ROI points.



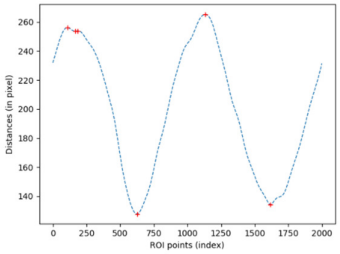
(b) Lesions border (green line), irregularity locations (red dots).



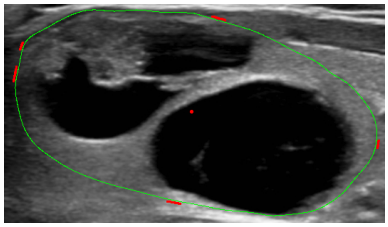

(c) Distance function of the irregular case with 90 interpolated points using Equi-angdisp sampling.



(d) Irregularity locations (red strips).



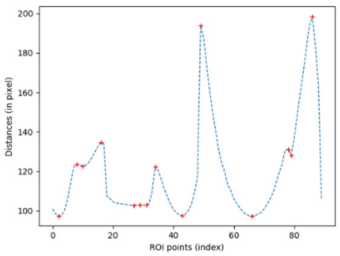

(e) Distance function of a regular case with 90 interpolated points using Equi-angdisp method.



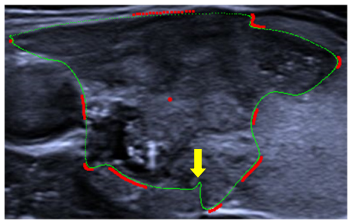
(f) Irregularity locations (red strips).

Figure 5.15: Visualization of borderline irregularity using distance function from the centroid.

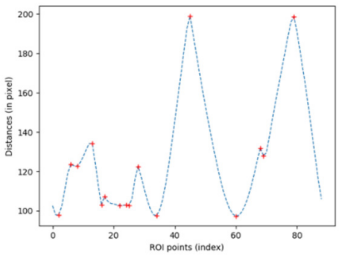
Figure 5.16 below shows the irregularity locations of another irregular case using Equi-angdisp and Equi-arclength sampling methods for comparison. Both border sampling techniques reveal similar irregular places, but the Equi-arclength method captures the location indicated by the yellow arrow in Figures b and d, while the Equi-angdisp method does not. The yellow arrow location shows obvious extreme turns in the borderline and can not be captured by the Equi-angdisp sampling method.



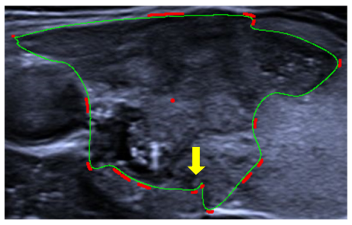

(a) Distance function using Equi-angdisp sampling.



(b) Irregularity visualization. (red stripes)



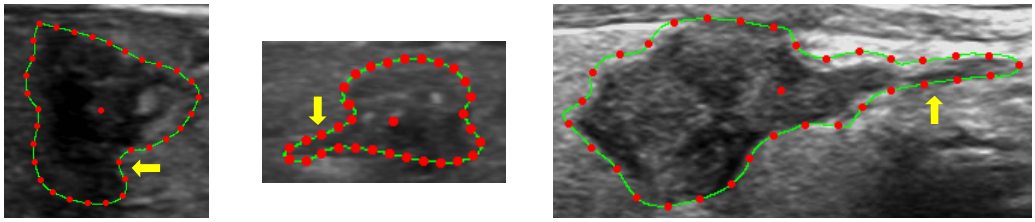

(c) Distance function using the Equi-arclength sampling.



(d) Lesions border (green line), irregular locations (red stripes)

Figure 5.16: Using the distances function to mark the minima and maxima and visualize the irregularity locations.

The simple measurements of the border distances to the centroid of the lesion might not be effective in all cases of thyroid cancer, especially those with extreme shapes that include protrusions and/or indentations of the border of the lesions. Examples of such lesions are illustrated in Figure 5.17 below, where the yellow arrows show the borderline's elongations, protrusions, and indentations. Detecting such anomalies is important when assessing border irregularity; therefore, we introduce further methods for building distance functions based on different reference shapes.



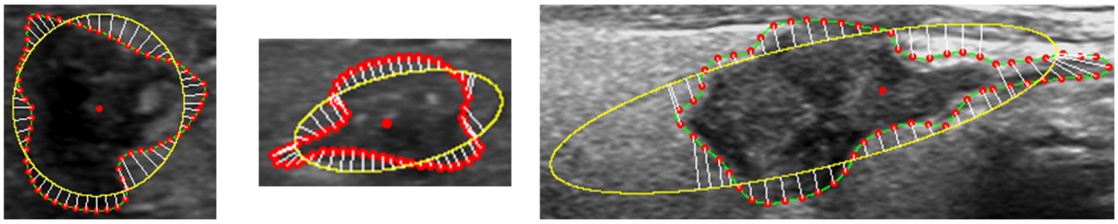
(a) (b) (c)

Figure 5.17: Some irregular cases of borders with elongations, protrusions, and indentations using Equi-arclength sampling method.

### 5.4.5 Distance Function to Fitted Ellipse

Since most cancer lesions have some elliptical shape, using a fitted ellipse of the ROI interpolated border points as a reference for distance measurements might help avoid the abovementioned problems and help detect the observed anomalies. It is anticipated that a fitted ellipse of the ROI border points will accurately reflect the overall shape of the lesion and hence, the level of border irregularity. Two methods are known for determining the fitted ellipse of any finite set of planar points: the least square errors and the Singular Value Decomposition (SVD) method [166]. SVD of a matrix is commonly used to solve general least square problems, including that of fitted lines and ellipses. An equation with

unknown coefficients is obtained for each point in the set of points used for fitting an ellipse forming a linear system of equations. The equations are arranged in a matrix and decomposed into S, V, and D matrices by the method of SVD and solved by minimizing algebraic distances [166]. Figure 5.18 shows three irregular cases along with their fitted ellipses (yellow curve) and measured distances (white lines) using the Equi-arclength sampling method with 64 (number of section n=6) sampled points.

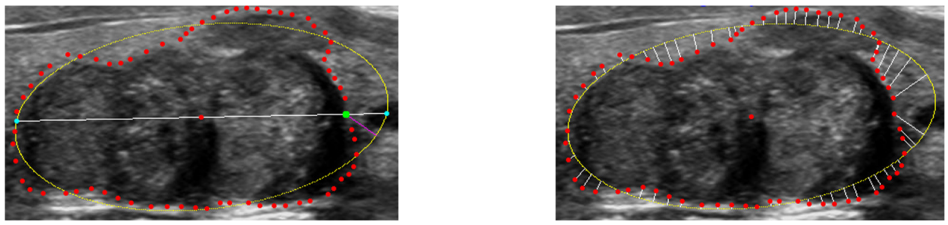


(a) (b) (c)

Figure 5.18: Using fitted ellipse as reference for building the distance function.

Computing the distance function depends on determining the Euclidian distance from each interpolated border sample point to their nearest points on the ellipse. The distance can be determined using a perpendicular line from the sample point to the ellipse's tangent. Implementing this approach need solving complicated quadratic equations. Instead, we use an iterative procedure by checking a series of points along the ellipse boundary for the one having the shortest distance to the lesion's border point. Figure 5.19 illustrates the procedure, starting with drawing a line from the border point (green dot Figure a) to the ellipse's centre (white line Figure a). Then, the two intersection points of the line with the ellipse boundary are marked (two light blue dots in Figure a). The distance from the border point to the nearest intersection point is measured where the iterative search for the shortest distance starts.

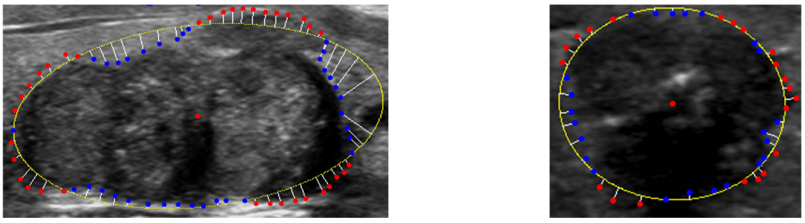
The iterative search examines the distances (purple line Figure a) of the neighbouring points of the intersection point (light blue dot) in both clockwise and anti-clockwise directions. When the distance of the nearby neighbouring points is greater than the actual distance, the search stops in that direction and continues in the opposite direction. The procedure ends when the distance from at least one of the neighbouring points in both directions is greater than the actual one. Figure b below illustrates the final measured distances as white lines.



(a) The shortest distance (purple line) (b) Final measured distances (white lines).

Figure 5.19: Distances from border points to the fitted ellipse for building distance function. Fitted ellipse (yellow curve), ROI points (red dots), measured distances (white lines).

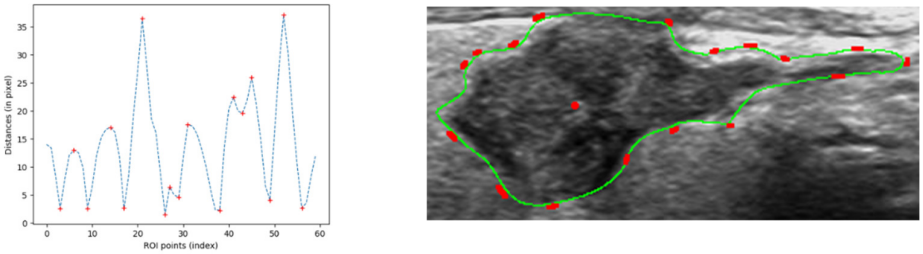
In opposition to the centroid as a reference, the fitted ellipse can be used to determine border shape abnormalities such as border protrusions and indentations. For that, each border point is examined for being inside or outside the fitted ellipse. The points outside the fitted ellipse represent border protrusions, while the points inside the ellipse represent the indentations. The more protrusions and indentations a nodule has, the more irregular the border is. The severity of the irregularities could also be measured by calculating the underlying areas between the protrusions/indentations and the fitted ellipse boundary. Figure 5.20 below shows two irregular cases of thyroid lesions showing the protrusion border points in red and the indentation points in blue. Six protrusions and indentations can be counted in the first and ten in the second, but they are much smaller in size. It might be more informative to give the distances different signs (negative or positive) depending on whether the actual border point lying inside or outside the fitted ellipse before it can be used to form the distances' function for the use as a feature vector for the input to a classifier.



(a) (b)

Figure 5.20: Visualization of border protrusions and indentations using fitted ellipse.

Finally, the distances function can be used in the same way as in the centroid-based distances to visualize irregular locations on the lesion border. Figure 5.21 in the following illustrates the irregularity visualizations of an irregular case.



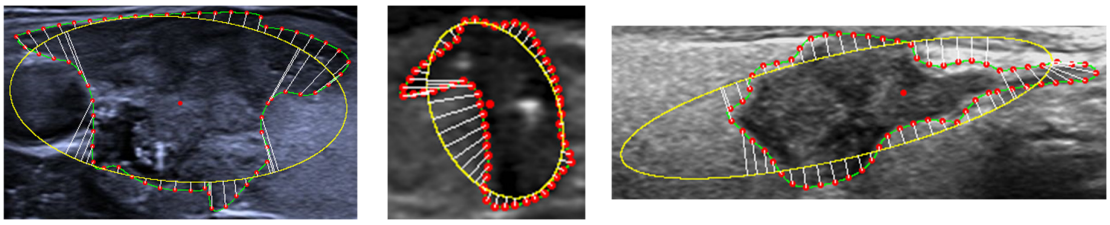

(a) (b)

Figure 5.21: Using distance function from fitted ellipse to visualize locations of the border irregularity of an irregular case.

### 5.4.6 Distance Function to Fitted Gaussian

Although the distance function obtained from the fitted ellipse seems more informative than the simple distances from the centroid, the ellipse may still not reflect the whole border irregularity of some lesions of extreme shapes like the ones in Figure 5.22. The distances (white arrows) do not fully reflect the

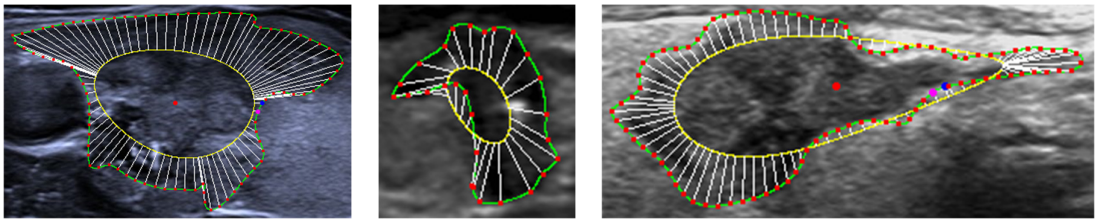
different lesions' elongation, protrusions and indentations shown. Therefore, a new Gaussian fitting method is attempted to improve the method's performance in capturing such border irregularities.



(a) (b) (c)

Figure 5.22: Fitted Ellipse as reference shape for some lesion cases of extreme shapes.

The new fitting method uses a Gaussian filter to interpolate the ROI points or the points from the interpolated border to build a Gaussian shape as a reference for the distance's measurement. The Gaussian fitted curve (yellow curve Figure 5.23) is a smoothly distorted ellipse nearer to egg shapes. The same iterative method for finding the shortest distance between the border points and reference shape, as described in the case of the fitted ellipse (see section 5.4.5), is used here to determine the distance function. Figure 5.23 below shows the Gaussian fitting method applied to the same cases as in the ellipse fitting shown in Figure 5.22 above. This way of determining distance is adequate for the majority of lesion shapes, but in some cases (Figures 5.22 b and c), measuring border distance at the points where the Gaussian shape border and the lesion border cross is problematic. Finally, the choice of the Gaussian filter sigma value is challenging and, in our case, is solved empirically by testing a range of values.



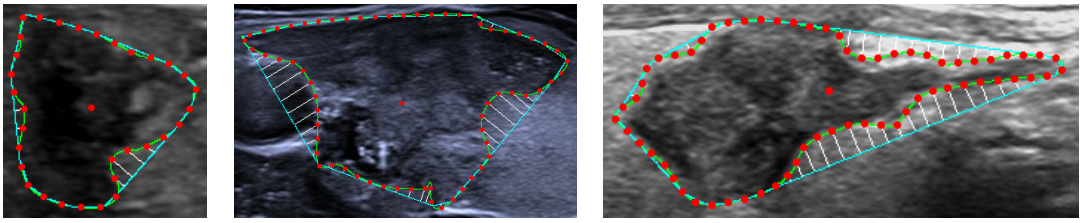
(a) (b) (c)

Figure 5.23: Fitted Gaussian method for some irregular cases of extreme shapes. Fitted Gaussian (yellow curve), border points (red dots), and Interpolated borderline (green curve).

### 5.4.7 Distance Function to a Convex Hull

Besides the two previously described shapes, we investigated using another regular reference shape corresponding to the convex hull of the ROI or border set of points. The convex hull is the most relevant morphological representation of the lesion shape beside the entire ROI point set and can be used as a reference for the distance measurement. The Convex hull of a set of points is the smallest convex polygon that contains all the points in its interior or boundary. It has many applications in mathematics, statistics, economics, and geometric modelling involving shape analyses [167]–[170] and the shape

complexity description of a cancer lesion using convexity [171]. Although the convex hull is rarely a regular polygon, we shall first adopt it as a polygonal reference shape for the distance measurement from the sampled border points for border irregularity recognition of thyroid nodules. The use of distances to the convex hull is a more realistic method than the distances to the bounding box edges that have been used in the literature [162]. The distance between any of the sampled interpolated border points and the convex hull can be obtained by computing their distances to the nearest convex hull edge. Figure 5.24 shows some regular and irregular cases, illustrating the distances to the convex hull (see white lines).

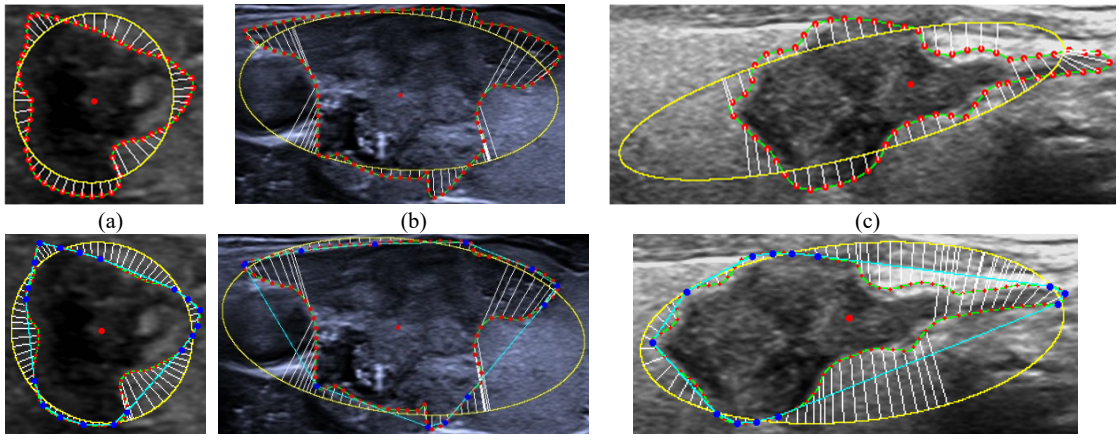


(a) (b) (c)

Figure 5.24: Convex hull (light blue curve) as reference shape for building distance function.

### 5.4.8 Distance Function to a Fitted Ellipse from Convex Hull

Besides the fitted ellipse of all the border points, we investigated using the fitted ellipse of a meaningful subset of the border set of points. In particular, we are interested in the subset of border points forming the lesion's Convex Hull corners. Fitting an ellipse on this subset of border points presents a more realistic ellipse shape, especially for elongation cases. Figure 5.25 below presents three irregular cases with both ellipse fitting on entire border points (Figures a, b, and c) and the subset of border points forming the convex hull (Figures d, e, and f) for comparison.



(a) (b) (c)

(d) (e) (f)

Figure 5.25: Comparison of the fitted ellipse of the full set of the border points and the fitted ellipse of the subset of border points forming the convex hull. Blue dots are the convex hull subset points.

The benefit of using the convex hull subset of border points for fitting the ellipse is more evident in the case shown in Figures c & f with elongation on the right side. Also, the number of peels and their centroid's trace of the convex hull could show the complexity of the lesion's shape and give indications about irregularities, which are described in one of the future works in the last chapter.

### 5.4.9 Feature Vector Representations of Distance Function

Classification of lesion borders using the various distance functions require the extraction of fixed-length feature vector representations of these 1-dimensional curves. An obvious way to do so is to take the distance function values at the fixed-size sample border points using one of the border point sampling methods. In this case, we need to use the same fixed number of sample points for each image. The cases depicted previously in Figures 5.13 and 5.14 show that for the same nodule, using different sample sizes results in visually different shapes for the corresponding distance function curve. Moreover, the shapes of the distance curves with the same sample size seem to distinguish between benign and malignant lesions. These illustrations show that adopting fixed-size sample distance functions is a sensible form of input to classifiers.

Using basic knowledge of the link between the shape of continuous single variable functions (the distance in this case) can be summarized by the number of its turning points, i.e., local maxima, local minima, and inflexion points. This could provide alternative feature vector representations of the various distance functions defined above. We shall briefly describe such representation in the Fast Fourier domain and concepts from the newly emerging Topological Data Analysis (TDA) paradigm applied to single-valued functions.

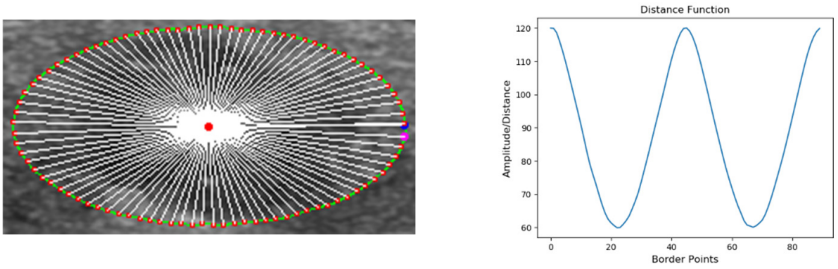
#### 5.4.9.1 FFT Feature Vector Representation

Fast Fourier Transformation (FFT) is a commonly used procedure in digital signal processing to analyse single/multiple valued input signal functions into its frequency domain consisting of original frequency components/sub-bands. For functions of a single real variable, FFT simply obtains a representation of the function as a linear (possibly infinite) combination of a set of sine and cosine functions of different amplitudes and frequencies. The principle of decomposing original signals into frequency components can be used in many other applications beyond signal processing, including computer vision. Many works have been reported using the FFT spectrum to determine the irregularity of the border of cancer lesions such as breast cancer [157], [158]. In our case, we attempt to use the distance function calculated from previously described methods as an input to the FFT for border irregularity recognition of thyroid nodules. The distance function can now be interpreted as a signal composed of different frequency components with different amplitudes. The various frequency components correspond to the high and low border irregularities, which include small zigzags and large protrusions and indentations along the lesion's border.

The distance function must be prepared before using as an input signal into FFT. To illustrate this, we use the simple distance function from the centroid for a synthetically generated perfect elliptic-shaped lesion (see Figure 5.26). First, the signal must be shifted to the centre (value 0 on the y-axis) so that its amplitude has symmetrical positive and negative parts (see Figure 5.27 a). This can be easily done by calculating the minimum (minAmpl) and maximum (maxAmpl) of the signal amplitude (in our case, distance function). Then, the whole signal (each distance value) is subtracted from the sum of the minAmpl and half of the height (maxAmpl – minAmpl) of the signal (see equation 5-1 below).

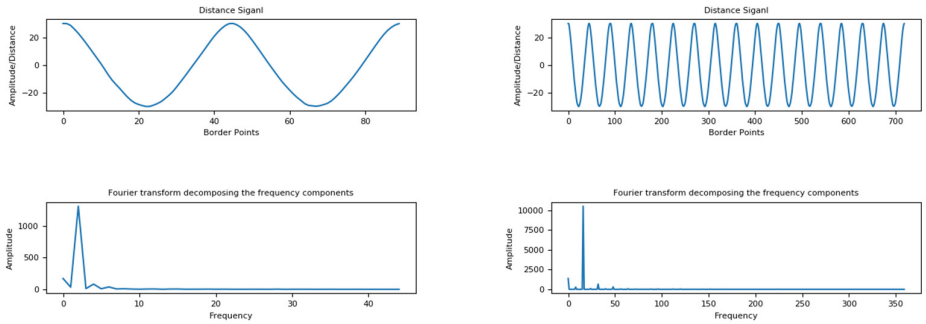
$$Signal = Signal - \left( minAmpl + \left( \frac{maxAmpl - minAmpl}{2} \right) \right) \qquad (5\text{-}1)$$

Next, for the FFT calculations to be more precise, the signal period is repeated several times (see the original signal Figure 5.27 a and repeated signal Figure 27 b). The signal is then converted into a frequency domain using FFT. Figures 5.27 a & b show the signal decomposed into frequency components (frequency spectrum) for both original and periodically repeated input signals. We can see from the figure the expected one frequency spike confirming that the original signal is composed of one sinusoidal signal since the original distance function of the perfect synthetic elliptic lesion has a sinusoidal form (see Figure 5.27 b).




(a) (b)

Figure 5.26: Synthetically generated perfect elliptical lesion and its distance function.

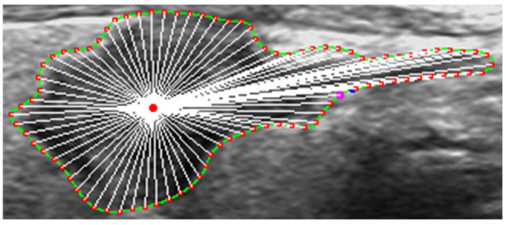



(a) Original not replicated signal. (b) Seven times periodically replicated signal.

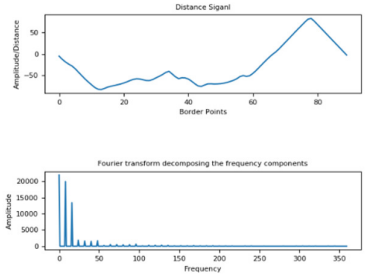
Figure 5.27: Pre-processing the distance function for input to FFT.

Figures 5.28 and 5.29 below illustrate the distance signals along with the FFT spectrums of several regular and irregular lesion cases, where the signals are replicated seven times before transforming into the FFT domain. The big jump in the distance signal for the case shown in Figure a represents the right-side elongation of the nodule and is reflected in the FFT spectrum through the two high amplitude spikes in the lower frequency range. The distance signal of the case in Figure c has almost a sinusoidal shape, as expected for a regular case; therefore, its frequency spectrum shows one high amplitude spike in the lower frequency. The distance signal of the case in Figure e shows several small and big jumps indicating a high irregular lesion reflected as several high amplitude spikes close to the low-frequency region. Finally, the last case shown in Figure g shows many small and few big jumps in the distance signal. Its frequency spectrum shows many high and middle amplitude spikes spread from the low to the high-frequency regions, indicating an irregular border, although the small zigzags along the border might not be considered an irregular case which may explain why the doctors labelled this case as regular.

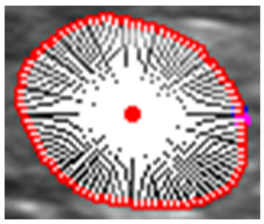
In the cases described above, we can observe that, in general, the more irregular the lesion is, the higher the number of its FFT spikes. The significant (or global) irregularities, such as elongations on the different sides of the lesion, are represented as high amplitude spikes in the lower frequencies. In contrast, the little zig-zag (local irregularity) along the borderline is exhibited by the spikes in the higher frequency. All these observations suggest that the FFT spectrum is a good feature for classifying regular and irregular nodule borders. The frequency spectrum of the distances function can be used either directly as a feature vector for input to a classifier or can be further processed for the search for discriminative features. These will be investigated in the coming sections.



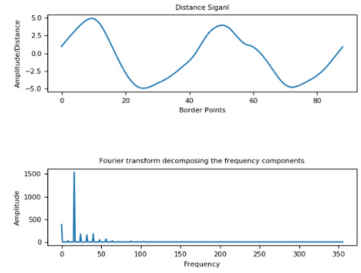
(a) Irregular case.




(b) FFT spectrum.

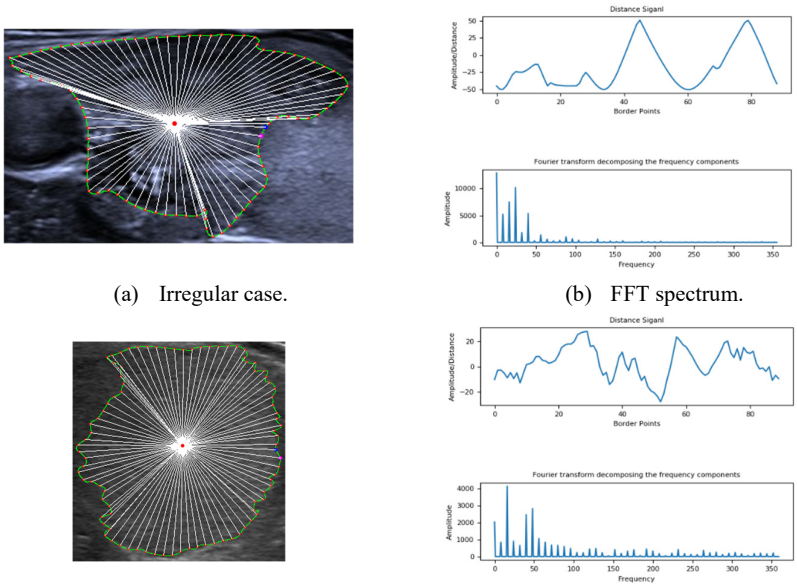


(c) Regular case.




(d) FFT spectrum.

Figure 5.28: FFT Spectrum of regular and irregular cases.



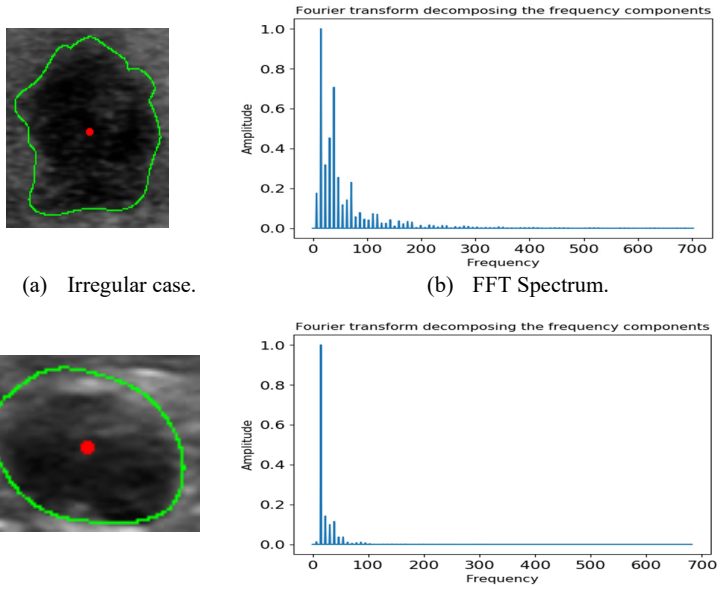

(a) Irregular case. (b) FFT spectrum.

(c) Regular case. (d) FFT spectrum.

Figure 5.29: FFT spectrum of several regular and irregular cases.

**The Number of Frequency Spikes as an Irregularity Indicator:**

As explained before, there is a strong correlation between the number of frequency spectrum spikes and their amplitudes and the border irregularity. Therefore, we investigate the number of spikes as a discrimination feature for border irregularity. As mentioned in the literature review (see section 5.2.2), some works have already reported using the FFT spectrum of distances function to extract irregularity features from lesion boundary; however, they are used for benign and malignant classification [157], [158]. The reported works use the FFT spectrum directly as a feature vector or calculate an irregularity index from the spectrum. In contrast, we intend to use the frequency spikes in each of the lesion's FFT spectrum as a feature vector. The high amount of tiny frequency spikes (see Figure 5.30) needs to be removed from the spectrum before counting the number of spikes, for instance, by using a threshold. Since the amplitude of the frequency spikes depends on the lesion size, there is a high amplitude variation across our dataset. Therefore, the spectrum amplitudes need to be normalized before extracting features by thresholding, e.g., by using normalization by range. The cases shown in Figure 5.30 include normalized FFT spectrum, where the frequency amplitudes always vary between 0 and 1. Then, the number of spikes can be counted by setting one or a range of amplitude thresholds. The number of spikes for each lesion case can be extracted by counting each spike having an amplitude greater than a threshold. For instance, using a range of thresholds from 0.02 to 0.2 generates ten spike numbers, which can be used as a feature vector of length ten.




(a) Irregular case. (b) FFT Spectrum.

(c) Regular case (d) FFT Spectrum.

Figure 5.30: Normalized FFT spectrum of some regular and irregular cases.

#### 5.4.9.2 TDA Feature Vector Representation

Topological Data Analysis (TDA) is a recently developed paradigm of big data analysis that is based on the assumption that data have shape. It provides a different insight than the data's other statistical or geometrical analyses. The so-called TDA tool of persistent homology (PH) represents the data shape by recording various connectivity parameters (connected components, non-contractable holes, non-contractible tunnels, …) in the shape when scanned at an increasing sequence of levels. For more details, see [172]. When applied to distance functions, or any function of a single variable (considered time series), the PH tool records the number of connected components (pieces of curves) seen up to each level when the curve is scanned vertically from the lowest point.

The distances measured from nodule border points (ROI points) to any of the different references (such as centroid, fitted ellipse, fitted ellipse from convex hull corners, convex hull, or fitted Gaussian) can be turned into distances function curves, interpreted as time series, for the topological data analyses. As two examples, distances from the centroid and fitted ellipse will be analysed using TDA, but using any of the other reference shapes is the same.

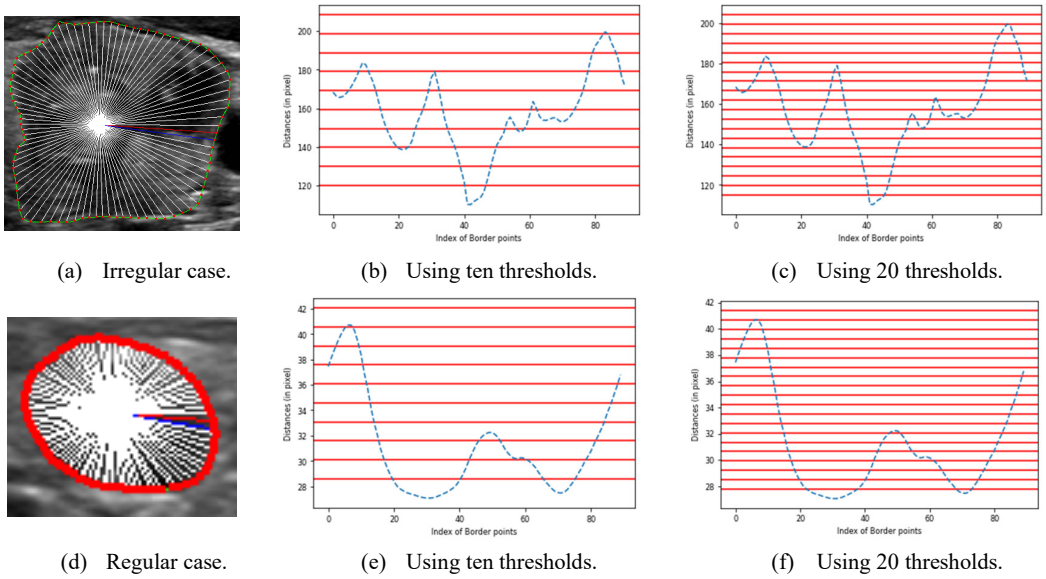
- **Distance Function from the Centroid**

Figure 5.31 shows the TDA analysis of the distance function of regular and irregular thyroid cancer cases. The Equi-arclength sampling method is used to select 90 points (red dots) from the interpolated border (green curve). The interpolated border distances (white lines) are measured from the centroid. The distance curve (blue dashed curve) is scanned using 10 (Figures b & e) and 20 (Figures c & f) distance thresholds for extracting TDA features. The thresholds are calculated by dividing the range

from the highest to lowest distance values by the number of thresholds or windows (see red lines). In the example shown in Figure b below, maximum distance =200, minimum distance =110, and the thresholds (red lines) can be calculated using the equation 5-2. The ten thresholds are: (120,130,140, 150, 160, 170, 180, 190, 200, 210).

$$\text{threshold} = maxDist + \frac{maxDist - minDist}{number\ of\ thresholds} \qquad (5\text{-}2)$$

The TDA analysis shown in Figure 5.31 starts with dividing the distance curve into equal horizontal regions (windows) using equally distributed amplitude thresholds (see the red horizontal lines). Then starting from the first threshold, i.e., from the first red line from the bottom, we count the number of connected curve pieces visible from the red line downwards. Repeating the process for each threshold produces one feature of the number of connected curve pieces. Thus using 10 and 20 thresholds produce feature vectors of lengths 10 and 20, respectively.




(a) Irregular case. (b) Using ten thresholds. (c) Using 20 thresholds.

(d) Regular case. (e) Using ten thresholds. (f) Using 20 thresholds.

Figure 5.31: TDA analysis using distances from the centroid in regular and irregular cases.

The extracted feature vectors (number of connected curve pieces) of both cases described above are presented in Table 5.4 below. In the irregular case shown in Figure a, using ten thresholds, only one piece of the connected curve can be seen downwards from the first threshold line; therefore, the number 1 will be recorded. One connected curve piece is still produced using the second threshold line, while two connected segments of the curve emerge at the third threshold line. The seventh threshold line reveals a maximum of 4 connected curve segments. Interestingly, a higher number of connected curve pieces can be observed in the irregular case compared to the regular one (see Table 5.4), which indicates

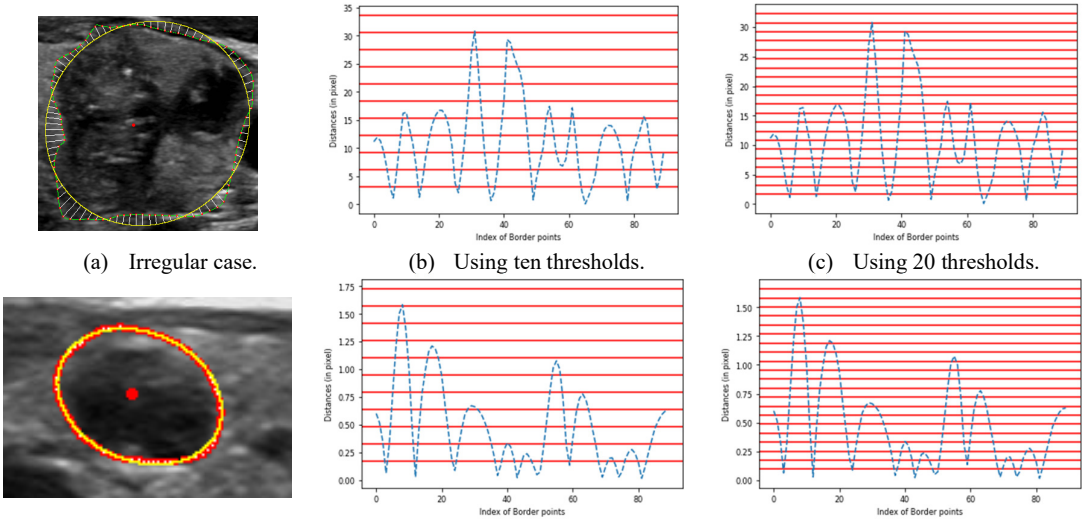
the discriminability of the TDA features for border irregularity. This correlation between the number of the connected curve pieces and the irregularity is still valid when 20 thresholds are used.

Table 5.4: Extracted TDA features using distance from centroid for regular and irregular cases.

| Lesion | TDA Feature vector (10 thresholds) | TDA Feature vector (20 thresholds) |
|---|---|---|
| Irregular | 1, 1, 2, 3, 3, 3, 4, 2, 2, 1 | 1, 1, 1, 1, 1, 1, 2, 2, 3, 3, 3, 3, 3, 4, 3, 2, 2, 2, 2, 1 |
| Regular | 2, 2, 2, 1, 1, 1, 2, 2, 2, 1 | 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 1 |

- **Distance Function from the Fitted Ellipse**

In the following (see Figure 5.32), we repeat the same procedure for TDA analyses of the distance curve using a fitted ellipse as a reference, whereby absolute distance values are used. We apply TDA to the same two cases used in the case of centroid reference described above for comparison. Generally, using both centroid and fitted ellipse as a reference show similar trends and correlations between the number of curve pieces and irregularity. However, the number of curve pieces in the case of the fitted ellipse is higher due to more fluctuations in the distance function. This behaviour might change if positive and negative values depending on whether the border points lying inside or outside the fitted ellipse are used to build the distance function. Surprisingly, the two regular and irregular cases using fitted ellipse show a very close number of curve pieces; however, their sequence pattern is different. We need to indicate here that the distances are not normalized; therefore, we can observe high fluctuations in the distance curve in the regular case shown in Figure d, although it has a very regular shape and distances range from 0 to 1.75 pixels. Table 5.5 shows the features extracted using the TDA.




(a) Irregular case. (b) Using ten thresholds. (c) Using 20 thresholds.

(d) Regular case. (e) Using ten thresholds. (f) Using 20 thresholds.

Figure 5.32: TDA analysis using distance from fitted ellipse for regular and irregular cases.

Table 5.5: Extracted TDA features using distance from centroid for regular and irregular cases.

| Lesion | TDA Features (10 thresholds) | TDA Features (20 thresholds) |
|---|---|---|
| Irregular | 8, 9, 9, 9, 5, 3, 3, 3, 2, 1 | 6, 8, 8, 8, 9, 9, 10, 9, 8, 7, 3, 3, 3, 3, 3, 3, 3, 2, 1 |
| Regular | 10, 6, 6, 5, 4, 4, 2, 2, 2, 1 | 9, 10, 8, 6, 6, 6, 6, 5, 5, 4, 4, 4, 3, 3, 2, 2, 2, 2, 2, 1 |

### 5.4.10 Method Inspired by Fractal Dimensions for Border Irregularity

Numerous studies have discussed the use of the FD to assess the roughness of a surface, a curve, or the textures of cancer lesions, including thyroid and other types of cancer, for discrimination of benign and malignant lesions (see sections 3.4.4 and 5.2.2). This motivated us to use an FD method to determine the irregularity of the thyroid nodule boundary. We first introduce the original FD method for measuring the roughness of boundaries and then introduce a method inspired by FD. As explained in section 3.4.4, the original FD method measures the boundary of a shape, such as a cancer lesion, at different scales; then, from the slope of the regression line of the log relation between the measured distances and the scales, the FD value is calculated, representing the roughness of the boundary. We investigated using the box-counting method for measuring FD of lesion border irregularity. For each image, one FD value is calculated as described in section 3.4.4 and used as a discrimination feature for the regular and irregular borders. However, the box-counting-based FD method of border irregularity did not give satisfactory results. Therefore, we introduced a new method inspired by FD for border irregularity recognition, described in the following.

The shortcoming of using the box-counting method to calculate the FD for the interpolated lesion border may be due to the bicubic spline model, which may result in the curves having higher frequencies in some places, especially for small lesions. We adopted a different approach for calculating a value (FDindex) representing the roughness or irregularity of the nodule boundary. The new method was inspired by comparing the interpolated nodule boundary perimeter with that of the regular fitted ellipse at different scales. First, we calculate the perimeter of the lesion by summing the Euclidian distances between every two successive points along the borderline. Measuring the perimeter at different scales is realized by choosing a different number of sampled points from the lesion boundary at equal distances using Equi-angdisp or Equi-arclength border point sampling methods (see sections 5.4.1.1 and 5.4.1.2). An irregularity index (FDindex) is calculated at each scale by dividing the lesion's perimeter by the ellipse's perimeter (see equations 5-3 below). Like the FD, the smaller the scale, the closer the calculated perimeter of the lesion to the true value. Assuming a better fit of the ellipse to the regular lesion boundary than the irregular one, the more regular the border, the closer the FDindex value to 1, and the more irregular the border, the further the value is from 1. The FDindexes can be used as an individual or a set of features for border irregularity classification for input to a chosen classifier.

$$\text{FDindex} = \frac{\text{interpolated border perimeter}}{\text{fitted ellipse perimeter}} \quad (5\text{-}3)$$

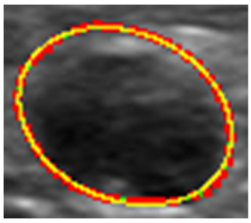
Many equations exist to calculate the perimeter of the fitted ellipse. In our case, we have used the following equation introduced by mathematician Ramanujan [173], which approximates the ellipse's perimeter.

$$h = \frac{(a-b)^2}{(a+b)^2} \quad (5\text{-}4)$$

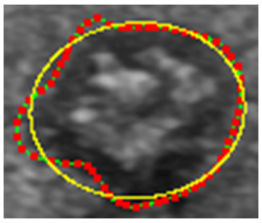
$$r \approx \pi(a+b)\left(1 + \frac{3h}{(10+\sqrt{4-3h})}\right) \quad (5\text{-}5)$$

Where h is a parameter calculated from major and minor axis a and b and r is the ellipse's perimeter.

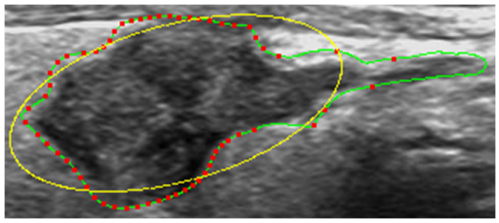
Using different scales for calculating the perimeter of the lesion gives slightly different FDindex values (see Figures 5.33 and 5.34). The two sampling methods for sampling interpolated border points did not show significant differences in the FDindex values; however, in the case of extreme shapes such as elongations, the Equi-arclength sampling method shows insufficient FDindex value under 1, which is due to the ellipse not fitting well in this case (see Figures in 5.34 c).



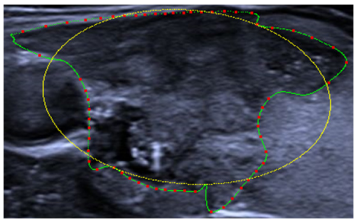
(a) FDIndex: 1.0111 Regular.



(b) FDIndex: 1.0588 Irregular.

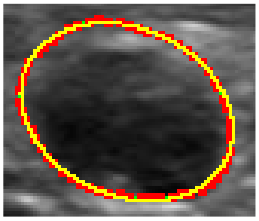


(c) FDIndex: 1.1147 Irregular.

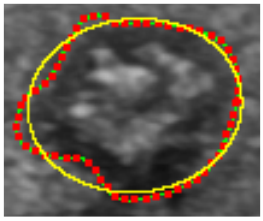


(d) FDIndex: 1.1852 Regular.

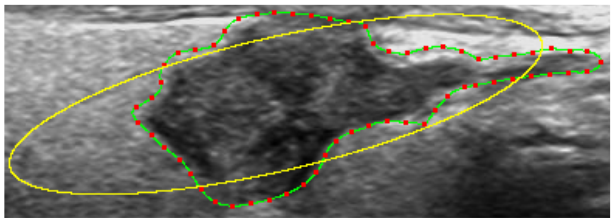
Figure 5.33: Fractal dimensions for calculating irregularity index FDindex. Sixty points were picked from the border using the Equi-angdisp sampling method.



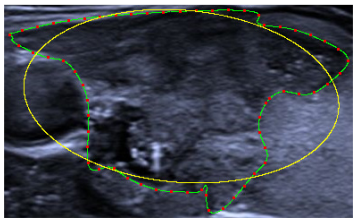
(a) FDIndex: 1.0029 Regular.



(b) FDIndex: 1.0548 Irregular.



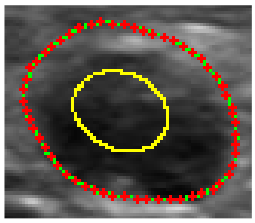
(c) FDIndex: 0.9348 Irregular.



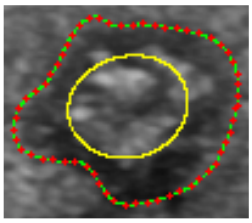
(d) FDIndex: 1.1632 Irregular.

Figure 5.34: FD-inspired method for calculating irregularity index FDindex. Sixty points were picked from the border using the Equi-arclength sampling method.

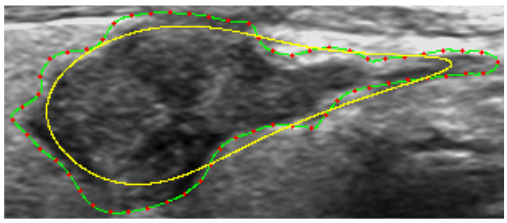
We also implemented a version of the FDindex method by using the Gaussian fitted curve and Equi-arclength sampling method instead of the fitted ellipse due to a better fit of the Gaussian shape, especially in the cases of extreme shapes (see Figure 5.35). However, due to the fixed sigma value, the FDindexes of the regular cases are higher than many other irregular shapes, as seen in Figure 5.35. For instance, the sigma value of 3 produces a small gaussian shape for the small lesion of a regular case (see Figure a), while a large size irregular lesion creates a large gaussian shape producing a smaller FDindex. In this case, to get realistic FDindexes using Gaussian shape, the sigma value needs to be dynamic and dependent on the size of the lesion. For this reason, we are conducting experiments on the FD-inspired method using only the fitted ellipse.



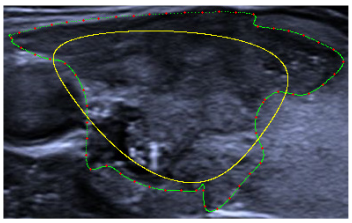
(a) FDIndex: 2.2384.



(b) FDIndex: 1.8608.



(c) FDIndex: 1.2385.



(d) FDIndex: 1.4705.

Figure 5.35: FDindex using Gaussian reference using fixed sigma value of 3 and 60 points were picked from the border using the Equi-arclength sampling method.

## 5.5 Experimental Setups and Results

In this section, we use the datasets described previously in section 5.3 to evaluate different proposed methods for thyroid cancer irregularity. We first describe the experimental setup, including the training and testing protocols and iterative classifier, before presenting and comparing the experimental results.

### 5.5.1 Experimental Setups

The general approach for testing and evaluating our methods is to use the bigger dataset of 395 images with one doctor's class label as a training and internal testing set and the smaller dataset of 100 images with three radiologists' class labels as an external testing set. The four potential protocols for evaluating the proposed methods are described in the following.

#### 5.5.1.1 Evaluation Protocols

We introduce the following four potential protocols for evaluating our proposed methods introduced in this chapter and chapter 6. The protocols aim to balance the training data or to produce one model for external deployment. Depending on the type of method certain protocol is used in the evaluations.

- **Evaluation Protocol1**

We used here the common evaluation protocol five-fold cross-validation that splits the dataset into training and testing sets k times or k folds, where in this case, k=5. The protocol takes 20% of the entire dataset for testing and the remaining 80% for training a model in each split. The data split is repeated five times (five-fold) to ensure that every image in the dataset is used once in the testing. The average accuracies, specificities, and sensitivities of the five folds represent the overall performance of the models. Although all five models could be used for testing the external dataset by averaging their accuracies, selecting one model for external testing is more convenient.

- **Evaluation Protocol2**

This protocol is similar to five-fold cross-validations; however, it balances the number of cases in each of the two classes used to train the models. The whole minor class (regular) and the same amount of randomly selected cases from the major class (irregular) are used to train the models in 5-fold cross-validation. Then, the balanced splitting process is repeated five times (5x), where in each iteration, five models of the five folds are generated, producing a total of 25 models. The average accuracies of the 25 models are used as the final performance measure. Because we are averaging the accuracies among 25 models, we face difficulty choosing the best model for testing the external dataset.

- **Evaluation Protocol3**

This new evaluation protocol aims to select one model that solves the issues of the previous two protocols for testing the external dataset. In this protocol, we first randomly select 30% (118 images) of the internal dataset DS(395) for internal testing. The remaining 70% (277 images) are randomly split

into training and evaluation sets in 80% and 20% ratios, respectively. The random training/evaluation splits are repeated 100 times, and the evaluation's accuracy, sensitivity, and specificity are recorded. From 100 trained and evaluated models, the top 20 models having the highest accuracies, the model with the smallest gap between sensitivity and specificity, is selected for evaluating the internal and external testing sets. The number of regular and irregular cases for training, evaluation, and testing are shown in Table 5.6 below. The flowchart shown in Figure 5.36 explains the steps for splitting the dataset or the feature vectors (FV) into different portions of training/evaluation and internal/external testing.

Table 5.6: Number of cases in training, evaluation, and internal testing (protocol3).

| Class | No. Images | Training& Evaluation | Testing |
|---|---|---|---|
| Regular | 181 | 127 (Random101T/26E) | 54 |
| Irregular | 214 | 150 (Random 120/30E) | 64 |
| Total | 395 | 277 | 118 |

- **Evaluation Protocol4**

This protocol is the same as protocol3, except it uses a balanced split of the regular and irregular classes for training and evaluation; the ratio of the internal testing set remains the same. From the remaining 277 cases for training/evaluation, we mix the whole minor class (regular) with the same amount of randomly selected images from the major class (irregular). Afterwards, the balanced image set or feature vector is randomly split into 80% for training and 20% for evaluation. The remaining procedure is the same as in protocol3. The selected model, as described before, is used for testing the internal and external sets. Table 5.7 below shows the class ratios for the training, evaluation, and testing.

Table 5.7: Number of cases in training, evaluation, and internal testing (protocol4).

| Class | No. Images | Training& Evaluation | Testing |
|---|---|---|---|
| Regular | 181 | 127 ➔127 (Random101T/26E) | 54 |
| Irregular | 214 | 150 ➔127 (Random 101/26E) | 64 |
| Total | 395 | 277 | 118 |

The different protocols described above have each their benefits; however, we decided to use protocol3 for our handcrafted feature-based methods introduced in this chapter and most of the methods in chapter 6 since it produces one model that can be easily used for testing the external dataset or deployed in a prototype for clinical trials. Since our dataset is nearly balanced, we did not deploy protocol4. For the sake of comparison, the image IDs of the internal testing and the 100 splits are fixed across all the experiments by setting fixed random seeds, i.e., the same images are used in internal testing and each of the 100 splits. On the other hand, training 100 CNN-based models (see chapter 6) on different dataset splits is time-consuming; therefore, we used the five-fold cross-validations (protocol1) in this case.

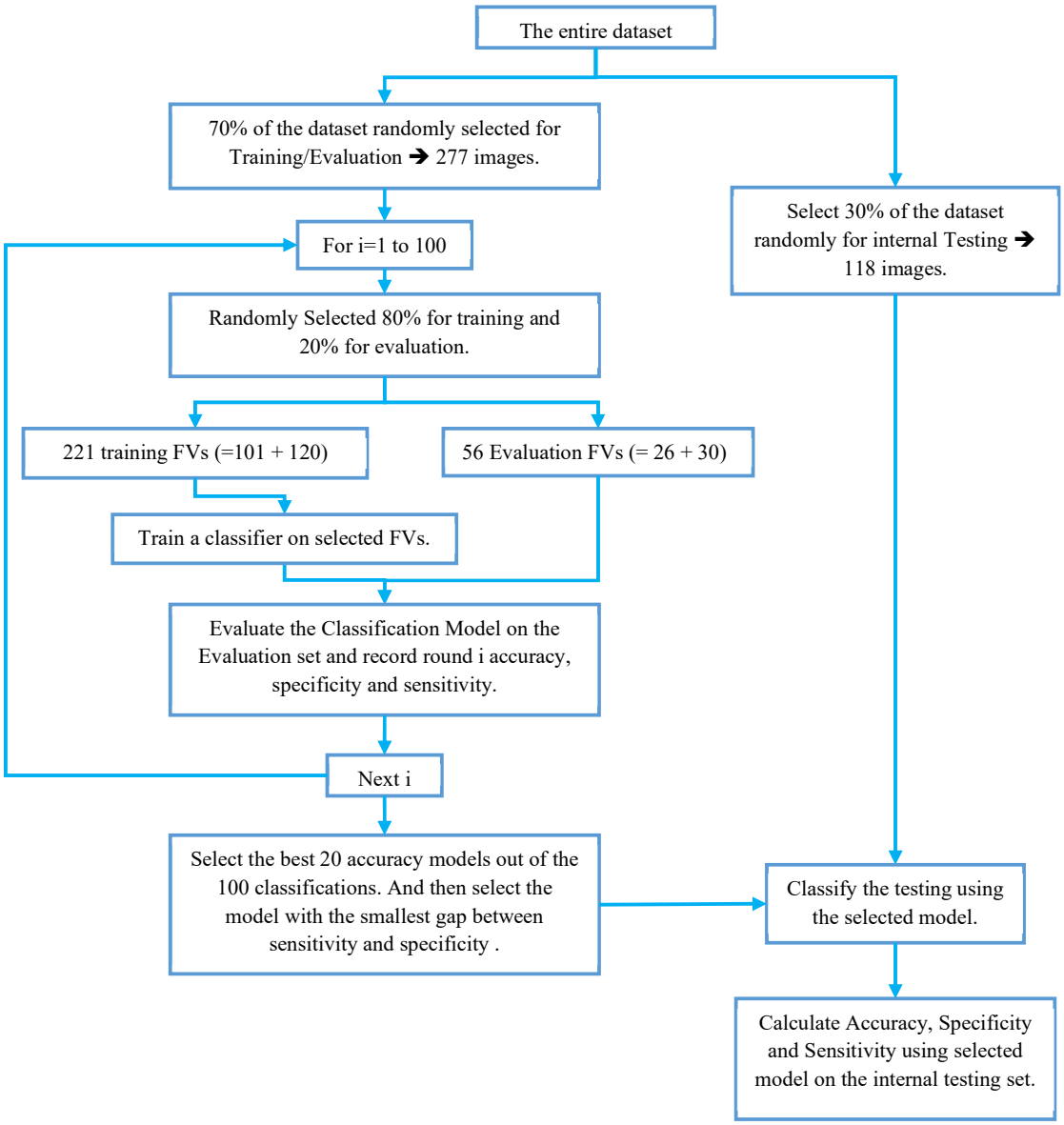



Figure 5.36: Flow chart of evaluation protocol3.

#### 5.5.1.2 Iterative Classifier

Evaluation protocol3 is used to assess the performance of most of the methods proposed in this work for border irregularity recognition. However, some methods produce single feature values, which do not perform well with conventional classifiers such as kNN or SVM. Therefore, a simple classifier called Iterative Classifier is developed that uses an ideal threshold of the single feature to split the entire dataset giving the highest accuracy. To find the ideal feature threshold, a range of feature values from the minimum to the maximum, determined over the entire dataset, is used as the threshold to split the dataset. For instance, the minimum and maximum feature value ranges are divided by 500, creating 500

thresholds. The dataset or its feature vectors is split using each threshold, and the accuracy, specificity, and sensitivity are recorded. Finally, from the top 20 highest accuracy thresholds, the one having the smallest gap between sensitivity and specificity is selected. The selected threshold can then be used to split the external testing set and calculate the accuracy, specificity, and sensitivity. The iterative classifier is used to test the performance of FD-inspired and FFT-based methods when single features are extracted for irregularity classification.

#### 5.5.1.3 Hardware Setup

The same hardware described in chapter 4 for crack recognition experiments is used in this part of the work. Contrary to the experiments in chapter 4, all hand-crafted and deep learning-based methods evaluations in this chapter and chapter 6 are run on the windows server with two GPUs and 128RAM.

### 5.5.2 Experimental Results

In this section, we present the results of our extensive experimental work to determine the performance of the different border distance-based and FD-based schemes. The range of experiments was much more extensive than presented here because, during the early days, we only had access to the first dataset provided by a hospital in China, whose images were labelled by one single clinician, with four different subclasses of regular and irregular cases. The results of those experiments are not presented here but are available in an accompanying folder to this. Those experiments were very useful in the stage of developing our schemes, but it became difficult to compare their performances on an external dataset without imposing variable index thresholding to reduce the problem into a binary classification. In general, the pattern of the results is not much different from the ones presented here. The presented results in this thesis correspond to training and testing using the more recently recorded dataset of 395 images labelled using 2-class regular and irregular labels, again by a single clinician who also selected the lesion's border ROI points (see section 5.3). For each scheme, we additionally test the performance of the model obtained with the adopted protocol3 (see section 5.5.1.1) on a dataset of 100 images labelled by three different clinicians. The section ends by comparing the performance of the different proposed schemes.

Regardless of the reference used, all distance-based approaches normalise the distances by dividing each value by the maximum distance between the border points and the centroid. The normalization step is necessary to ensure that distances between nodules of varied sizes in the dataset always range between 0 and 1. In all experiments, the average accuracy (**Acc**), specificity (**Reg**), and sensitivity (**Irreg**) on internal and external testing sets are recorded. The external testing results include the individual class labels of the three doctors and the subset where the three doctors agree on the label. The results are repeated using different amounts of sampled points from the interpolated border to determine the one giving the best performance. Both kNN and SVM classifiers are attempted for the

various experiments. However, the SVM gave overall higher performances and is used in all the experiments presented in this chapter.

#### 5.5.2.1 Distances Function from the Centroid

The following results present the performance of the simple distances from the centroid to the border points for irregularity classifications. Here, the distance function is used as a feature vector and fed to the classifier for classification. Both Equi-angdisp and Equi-arclength sampling methods are used to select interpolated border points for comparison.

- **Equi-angdisp Sampling Method**

Table 5.8 below shows the results of distance based method using centroid as a reference and Equi-angdisp as a sampling method. The results on internal and external sets are presented in each row, showing the three doctors' individual and agreed class labels for the external tests. The experiments are done using different numbers of sampled points from the interpolated border (**NoPnts**). The row in bold was selected as the best-performing model with the 135 sampled points.

Table 5.8: Experimental results of the distances-based method using centroid. (Equi-angdisp sampling).

| **Distances (Centroid)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 75 | 74 | 75 | 69 | 64 | 73 | 71 | 68 | 73 | 74 | 67 | 80 | 77 | 67 | 84 |
| 15 | 75 | 70 | 80 | 78 | 70 | 84 | 78 | 72 | 82 | 79 | 69 | 88 | 86 | 73 | 95 |
| 25 | 77 | 76 | 78 | 72 | 61 | 80 | 74 | 65 | 80 | 75 | 63 | 86 | 79 | 67 | 88 |
| 30 | 81 | 85 | 77 | 77 | 70 | 82 | 75 | 70 | 78 | 78 | 69 | 86 | 84 | 73 | 91 |
| 45 | 81 | 81 | 80 | 76 | 70 | 80 | 74 | 70 | 77 | 77 | 69 | 84 | 82 | 73 | 88 |
| 90 | 78 | 76 | 80 | 73 | 64 | 80 | 75 | 68 | 80 | 76 | 65 | 86 | 81 | 70 | 88 |
| **135** | **82** | **83** | **81** | **75** | **70** | **79** | **73** | **70** | **75** | **76** | **69** | **82** | **81** | **73** | **86** |
| 190 | 79 | 78 | 80 | 73 | 64 | 80 | 75 | 68 | 80 | 76 | 65 | 86 | 81 | 70 | 88 |

- **Equi-arclength Sampling Method**

Since we use the Equi-arclength border point selection method, we can not select the same range of border points used in the above experiment. Equi-arclength sampling method always takes the number of points = $2^n$ Where n is the number of bisections used to select border points.

Number of border points = $2^n$

The results in table 5.9 demonstrate deteriorated method performances when the Equi-arclength sampling method is used rather than the Equi-angdisp sampling. However, this behaviour is reversed when the method is tested on the external testing set for three doctors' class labels as well as for the agreed cases. Even though the second sampling method (Equi-arclength) visually samples the lesion

boundary better, in some datasets, it may not capture border irregularity as good as the first sampling method (Equi-angdisp).

Table 5.9: Experimental results of the distances-based method using centroid. (Equi-arclength sampling)

| **Distances (Centroid)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 8 | 75 | 81 | 70 | 73 | 75 | 71 | 73 | 78 | 70 | 80 | 80 | 80 | 82 | 83 | 81 |
| 16 | 75 | 78 | 73 | 78 | 82 | 75 | 78 | 85 | 73 | 83 | 84 | 82 | 88 | 93 | 84 |
| 32 | 74 | 78 | 70 | 79 | 82 | 77 | 79 | 85 | 75 | 84 | 84 | 84 | 89 | 93 | 86 |
| 64 | 74 | 78 | 70 | 79 | 82 | 77 | 79 | 85 | 75 | 84 | 84 | 84 | 89 | 93 | 86 |
| **128** | **76** | **83** | **70** | **75** | **75** | **75** | **75** | **78** | **73** | **82** | **80** | **84** | **85** | **83** | **86** |
| 256 | 74 | 78 | 70 | 79 | 82 | 77 | 79 | 85 | 75 | 84 | 84 | 84 | 89 | 93 | 86 |

### 5.5.2.2 Distances Function from Fitted Ellipse

The experimental results of the distances-based method using the fitted ellipse as a reference shape are presented in the following. In contrast to other reference shapes, the distances can have both positive and negative values depending on whether the border points are inside or outside the fitted ellipse.

- **Equi-angdisp Sampling Method**

The performance of the method on the internal testing is slightly lower than the method using the centroid presented in the previous section (see Table 5.10). However, the method performance is slightly better on the external testing sets. Its performance on external agreed cases is lower than the method based on centroid. In this case, the fitted ellipse as a distance reference does not demonstrate advantages over the centroid.

Table 5.10: Experimental results of the distances-based method using fitted ellipse. (Equi-angdisp sampling).

| **Distances (Fitted Ellipse)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| **One Doctor** | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 76 | 80 | 73 | 79 | 77 | 80 | 71 | 70 | 72 | 78 | 73 | 82 | 82 | 77 | 86 |
| 15 | 76 | 76 | 77 | 75 | 73 | 77 | 71 | 70 | 72 | 76 | 71 | 80 | 81 | 77 | 84 |
| **25** | **78** | **83** | **73** | **74** | **77** | **71** | **72** | **78** | **68** | **77** | **78** | **76** | **81** | **83** | **79** |
| 30 | 74 | 80 | 69 | 71 | 77 | 66 | 71 | 80 | 65 | 74 | 78 | 71 | 78 | 83 | 74 |
| 45 | 70 | 80 | 62 | 73 | 80 | 68 | 71 | 80 | 65 | 76 | 80 | 73 | 79 | 83 | 77 |
| 90 | 72 | 80 | 66 | 72 | 80 | 66 | 68 | 78 | 62 | 73 | 78 | 69 | 77 | 80 | 74 |
| 135 | 72 | 80 | 66 | 74 | 80 | 70 | 70 | 78 | 65 | 75 | 78 | 73 | 79 | 80 | 79 |
| 190 | 72 | 80 | 66 | 74 | 80 | 70 | 70 | 78 | 65 | 75 | 78 | 73 | 79 | 80 | 79 |

- **Equi-arclength Sampling Method**

The method's performance using the Equi-arclength sampling method is similar to the Equi-angdisp sampling method presented above (see Table 5.11). Overall, the performance of the fitted ellipse-based method is lower than the previous centroid-based method.

Table 5.11: Experimental results of the distances-based method using fitted ellipse. (Equi-arclength sampling).

| **Distances (Fitted Ellipse)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| **One Doctor** | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 8 | 70 | 74 | 67 | 63 | 61 | 64 | 65 | 65 | 65 | 70 | 67 | 73 | 71 | 67 | 74 |
| 16 | 72 | 78 | 67 | 73 | 75 | 71 | 73 | 78 | 70 | 76 | 76 | 76 | 81 | 80 | 81 |
| 32 | 73 | 78 | 69 | 70 | 80 | 62 | 70 | 82 | 62 | 79 | 86 | 73 | 79 | 83 | 77 |
| 64 | 72 | 74 | 70 | 70 | 77 | 64 | 70 | 80 | 63 | 79 | 84 | 75 | 79 | 80 | 79 |
| 128 | 71 | 74 | 69 | 71 | 77 | 66 | 71 | 80 | 65 | 78 | 82 | 75 | 79 | 80 | 79 |
| **256** | **75** | **78** | **72** | **71** | **77** | **66** | **73** | **82** | **67** | **78** | **82** | **75** | **81** | **83** | **79** |

#### 5.5.2.3 Distances function from Fitted Gaussian

In the following, the experimental results of the distance-based method using the Gaussian shape as a reference are presented. In this method, only positive distances from the Gaussian shape are used since the Gaussian shape rarely intersects the borderline of the nodule.

- **Equi-angdisp Sampling Method**

Table 5.12 shows results on internal, external, and agreed cases of the external testing set using the Equi-angdisp sampling method. Using 90 border points gives the best accuracy on internal testing; however, the best performance on external testing is achieved using 25 interpolated border points.

Table 5.12: Experimental results of the distances-based method using Gaussian shape as reference. (Equi-angdisp sampling)

| **Distances (Gaussian)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 73 | 72 | 73 | 74 | 64 | 82 | 72 | 62 | 78 | 75 | 63 | 86 | 79 | 67 | 88 |
| 15 | 75 | 78 | 73 | 75 | 68 | 80 | 73 | 68 | 77 | 82 | 73 | 90 | 82 | 70 | 91 |
| 25 | 76 | 74 | 78 | 77 | 68 | 84 | 77 | 70 | 82 | 82 | 71 | 92 | 86 | 77 | 93 |
| 30 | 74 | 72 | 75 | 78 | 70 | 84 | 76 | 70 | 80 | 81 | 71 | 90 | 86 | 77 | 93 |
| 45 | 75 | 74 | 75 | 77 | 68 | 84 | 75 | 68 | 80 | 82 | 71 | 92 | 76 | 76 | 77 |
| **90** | **78** | **76** | **80** | **75** | **66** | **82** | **73** | **65** | **78** | **80** | **69** | **90** | **76** | **78** | **75** |
| 135 | 75 | 74 | 75 | 76 | 66 | 84 | 74 | 65 | 80 | 81 | 69 | 92 | 80 | 80 | 80 |
| 190 | 76 | 74 | 78 | 75 | 66 | 82 | 75 | 68 | 80 | 80 | 69 | 90 | 78 | 78 | 78 |

- **Equi-arclength Sampling Method**

Table 5.13 shows results using the Equi-arclength sampling method. The performance is better than the Equi-angdisp sampling on internal testing and similar on external testing. The results show Gaussian shape captures the irregularity better than the fitted ellipse; however, it is not performing as well as the centroid-based method. We must remember that the fixed sigma value for constructing the Gaussian shape used here is not a good approach, and it might be replaced with a dynamic sigma value derived from the size of the lesion.

Table 5.13: Experimental results of the distances-based method using Gaussian shape as reference. (Equi-arclength sampling)

| **Distances (Gaussian)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| **One Doctor** | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 8 | 76 | 76 | 77 | 70 | 61 | 77 | 74 | 68 | 78 | 77 | 67 | 86 | 79 | 73 | 84 |
| 16 | 76 | 78 | 75 | 73 | 64 | 80 | 71 | 62 | 77 | 76 | 65 | 86 | 79 | 67 | 88 |
| **32** | **80** | **80** | **80** | **73** | **68** | **77** | **71** | **68** | **73** | **76** | **69** | **82** | **79** | **73** | **84** |
| 64 | 78 | 78 | 78 | 75 | 70 | 79 | 71 | 68 | 73 | 78 | 71 | 84 | 81 | 73 | 86 |
| 128 | 78 | 80 | 77 | 79 | 73 | 84 | 73 | 68 | 77 | 80 | 71 | 88 | 85 | 77 | 91 |
| 256 | 78 | 80 | 77 | 79 | 73 | 84 | 73 | 68 | 77 | 80 | 71 | 88 | 85 | 77 | 91 |

#### 5.5.2.4 Distances function from Convex Hull

The followings present the experimental results of the distance-based method using a Convex hull as a reference. The nodule border is not intersecting the outer peel of the convex hull formed from border points; therefore, the distances are always positive. As a reminder, the same normalization procedure is used here as in all other distance-based methods.

- **Equi-angdisp Sampling Method**

Table 5.14 shows the results using the Equi-angdisp sampling method. The method performance on the internal testing set is the highest among all previously presented methods, with an accuracy of 84%. However, its performance is lower than the centroid-based method on the external testing set.

Table 5.14: Experimental results of the distances-based method using Gaussian shape. (Equi-angdisp sampling)

| **Distances (Convex)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 81 | 91 | 73 | 73 | 86 | 62 | 69 | 85 | 58 | 78 | 88 | 69 | 81 | 90 | 74 |
| 15 | 79 | 89 | 70 | 72 | 89 | 59 | 70 | 90 | 57 | 81 | 94 | 69 | 81 | 90 | 74 |
| 25 | 82 | 93 | 73 | 74 | 89 | 62 | 74 | 92 | 62 | 83 | 94 | 73 | 85 | 93 | 79 |
| **30** | **84** | **93** | **77** | **75** | **89** | **64** | **73** | **90** | **62** | **84** | **94** | **75** | **85** | **93** | **79** |
| 45 | 81 | 91 | 73 | 75 | 89 | 64 | 75 | 92 | 63 | 84 | 94 | 75 | 86 | 93 | 81 |
| 90 | 82 | 91 | 75 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| 135 | 82 | 91 | 75 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| 190 | 81 | 91 | 73 | 76 | 91 | 64 | 74 | 92 | 62 | 85 | 96 | 75 | 86 | 93 | 81 |

- **Equi-arclength Sampling Method**

The following table shows the results of the method using the Equi-arclength sampling method. The method's performance in internal testing was identical to that of the Equi-angdisp method, but it performed better in external testing, reaching an accuracy of 88% on the agreed class labels. The Convex Hull-based method performs better than all prior methods when tested internally and is equivalent to the centroid method when tested externally. Despite this, the method is simple and requires no parameter settings.

Table 5.15: Experimental results of the distances-based method using Convex hul. (Equi-arclength sampling)

| **Distances (Convex)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 8 | 81 | 85 | 77 | 77 | 80 | 75 | 71 | 75 | 68 | 82 | 82 | 82 | 85 | 83 | 86 |
| 16 | 82 | 87 | 78 | 72 | 82 | 64 | 74 | 88 | 65 | 81 | 88 | 75 | 84 | 87 | 81 |
| 32 | 82 | 89 | 77 | 76 | 86 | 68 | 74 | 88 | 65 | 85 | 92 | 78 | 86 | 90 | 84 |
| 64 | 82 | 89 | 77 | 76 | 84 | 70 | 76 | 88 | 68 | 85 | 90 | 80 | 88 | 90 | 86 |
| 128 | 81 | 89 | 75 | 74 | 84 | 66 | 74 | 88 | 65 | 83 | 90 | 76 | 85 | 90 | 81 |
| **256** | **84** | **91** | **78** | **77** | **84** | **71** | **75** | **85** | **68** | **84** | **88** | **80** | **88** | **90** | **86** |

#### 5.5.2.5 Distances Function from Fitted Ellipse from Convex Hull

The followings present the results of the distances-based method using an ellipse fitted on the subset of corner points forming the convex hull of the border points. Although we tried using only absolute values of the distances, using both negative and positive distances from the ellipse gave better results and hence used in our experiments.

- **Equi-angdisp Sampling Method**

The method shows significant improvements ab to 5% on internal testing compared to the ellipse fitted to all border points presented before when Equi-angdisp sampling is used. The improvements are around 8% on the external testing set, reaching the highest accuracy of 90% on the agreed class labels (see Table 5.16).

Table 5.16: Experimental results of the distances-based method using fitted ellipse of the convex hull. (Equi-angdisp sampling).

| **Distances (Fitted Ellipse on Convex Hull)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 72 | 70 | 73 | 75 | 75 | 75 | 69 | 70 | 68 | 78 | 76 | 80 | 81 | 73 | 86 |
| **15** | **83** | **80** | **86** | **76** | **77** | **75** | **74** | **78** | **72** | **83** | **82** | **84** | **85** | **80** | **88** |
| 25 | 78 | 76 | 80 | 81 | 84 | 79 | 79 | 85 | 75 | 86 | 86 | 86 | 90 | 90 | 91 |
| 30 | 82 | 80 | 84 | 77 | 80 | 75 | 75 | 80 | 72 | 84 | 84 | 84 | 86 | 83 | 88 |
| 45 | 81 | 80 | 81 | 77 | 80 | 75 | 75 | 80 | 72 | 84 | 84 | 84 | 86 | 83 | 88 |
| 90 | 81 | 80 | 83 | 78 | 80 | 77 | 76 | 80 | 73 | 83 | 82 | 84 | 86 | 83 | 88 |
| 135 | 82 | 81 | 83 | 78 | 80 | 77 | 76 | 80 | 73 | 83 | 82 | 84 | 86 | 83 | 88 |
| 190 | 72 | 80 | 66 | 71 | 57 | 82 | 65 | 50 | 75 | 74 | 59 | 88 | 74 | 50 | 91 |

- **Equi-arclength Sampling Method**

Table 5.17 shows the results of the method when Equi-arclength sampling method is used. The performance is slightly lower on the internal testing than in the previous sampling method; however, the performance on the external testing is higher, reaching 92% on the agreed cases.

Compared to previous methods based on centroid, fitted ellipse, and Gaussian using an ellipse fitted on the outer peel of a convex hull as a reference significantly enhances the performance in terms of the

highest accuracy and the smallest gap between sensitivity and specificity. While the method outperforms the simple convex hull method on internal testing, its external test performances are superior.

Table 5.17: Experimental results of the distances-based method using fitted ellipse. (Equi-arclength sampling)

| **Distances (Fitted Ellipse from Convex Hull)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 8 | 75 | 78 | 73 | 76 | 77 | 75 | 70 | 72 | 68 | 81 | 80 | 82 | 82 | 80 | 84 |
| 16 | 75 | 78 | 73 | 78 | 80 | 77 | 74 | 78 | 72 | 85 | 84 | 86 | 86 | 83 | 88 |
| 32 | 78 | 76 | 80 | 81 | 86 | 77 | 79 | 88 | 73 | 88 | 90 | 86 | 92 | 93 | 91 |
| 64 | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| 128 | 80 | 76 | 83 | 80 | 82 | 79 | 76 | 80 | 73 | 87 | 86 | 88 | 89 | 87 | 91 |
| **256** | **80** | **81** | **78** | **79** | **86** | **73** | **75** | **85** | **68** | **86** | **90** | **82** | **88** | **93** | **84** |

#### 5.5.2.6 TDA Features

TDA analysis can be performed using any of the previously described distance functions; however, we only test the method with centroid and convex hull-based distance functions in order to test it with high- and low-performing distance functions and to limit the scope of the conducted experiments. The number of thresholds (**NoThresh**) used to capture the curve segment counts are: 10, 25, 50, 75, and 100. The values of curve segment counts are used to form a feature vector for input to a classifier. Contrary to the previous experiments, we present the results using only one set of the sampled points from the interpolated border to build the distance function.

- **Equi-angdisp Sampling Method (Distance from Centroid)**

Distances from the centroid, Equi-angdisp sampling, and 90 sampled border points are used in the experiments presented in Table 5.18 below using TDA analysis. It shows lower performance than other previously presented methods on internal and external testing sets.

Table 5.18: Experimental results of the distances-based method using centroid as reference using 90 border points and a different number of thresholds. (Equi-angdisp sampling)

| **TDA Distances (Centroid)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoThresh | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 74 | 69 | 78 | 71 | 64 | 77 | 65 | 57 | 70 | 70 | 61 | 78 | 74 | 67 | 79 |
| **25** | **74** | **72** | **75** | **72** | **66** | **77** | **68** | **62** | **72** | **73** | **65** | **80** | **77** | **73** | **79** |
| 50 | 74 | 69 | 78 | 71 | 64 | 77 | 69 | 62 | 73 | 72 | 63 | 80 | 77 | 70 | 81 |
| 75 | 72 | 72 | 72 | 72 | 66 | 77 | 68 | 62 | 72 | 73 | 65 | 80 | 77 | 73 | 79 |
| 100 | 74 | 70 | 77 | 74 | 70 | 77 | 68 | 65 | 70 | 73 | 67 | 78 | 78 | 77 | 79 |

- **Equi-arclength Sampling Method (Distance from Centroid)**

The results in Table 5.19 below show the results of the TDA method using Equi-arclength sampling. The number of selected border points is fixed at **128** border points using seven sections ($2^7$). The results show lower accuracies than using the Equi-angdisp sampling method on internal and external testing sets.

Table 5.19: Experimental results of the distances-based method using centroid as reference using 128 border points and different thresholds. (Equi-arclength sampling)

| **TDA Distances (Centroid)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoThresh | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 71 | 69 | 73 | 71 | 61 | 79 | 67 | 57 | 73 | 68 | 57 | 78 | 74 | 63 | 81 |
| 25 | 70 | 69 | 72 | 67 | 59 | 73 | 69 | 62 | 73 | 70 | 61 | 78 | 74 | 67 | 79 |
| **50** | **72** | **69** | **75** | **68** | **61** | **73** | **70** | **65** | **73** | **71** | **63** | **78** | **75** | **70** | **79** |
| 75 | 71 | 70 | 72 | 69 | 64 | 73 | 71 | 68 | 73 | 70 | 63 | 76 | 75 | 73 | 77 |
| 100 | 71 | 69 | 73 | 68 | 61 | 73 | 70 | 65 | 73 | 71 | 63 | 78 | 75 | 70 | 79 |

- **Equi-angdisp Sampling Method (Distance from Convex hull)**

The convex hull is used as a reference for building the distance function with the same experimental setting as before. The performance improved significantly compared to the method using a centroid as a reference, as seen in the Table below.

Table 5.20: Experimental results of the distances-based method using convex shape as reference using 90 border points and a different number of thresholds. (Equi-angdisp sampling)

| **TDA Distances (Convex)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoThresh | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 81 | 81 | 80 | 78 | 73 | 82 | 72 | 68 | 75 | 81 | 73 | 88 | 84 | 73 | 91 |
| 25 | 80 | 81 | 78 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| **50** | **81** | **81** | **80** | **82** | **82** | **82** | **74** | **75** | **73** | **81** | **78** | **84** | **88** | **83** | **91** |
| 75 | 81 | 85 | 78 | 82 | 82 | 82 | 76 | 78 | 75 | 83 | 80 | 86 | 89 | 87 | 91 |
| 100 | 81 | 85 | 78 | 82 | 84 | 80 | 74 | 78 | 72 | 81 | 80 | 82 | 88 | 87 | 88 |

- **Equi-arclength Sampling Method (Distance from Convex Hull)**

The experiment is conducted using Equi-arclength and distances from the convex hull with the same settings as before. As presented in Table 5.21, the method performs significantly better than the method when the Equi-angdisp sampling method is used on the internal testing set. However, it achieves slightly lower performance when tested on the external testing set.

The results show that using TDA to extract discrimination features from the distance function is valid, but its performance highly relies on how well the distance function is constructed. The performance is lower than all other methods previously introduced when the distance function from the centroid is

used. In contrast, its performance increases significantly when the distance function from the convex hull is used, outperforming all other previously presented methods on the internal testing set, reaching maximum accuracy of 87%.

Table 5.21: Experimental results of the distances-based method using convex shape as reference using 128 border points and different thresholds. (Equi-arclength sampling)

| **TDA Distances (Convex)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoThresh | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 86 | 89 | 84 | 79 | 73 | 84 | 71 | 65 | 75 | 78 | 69 | 86 | 84 | 73 | 91 |
| 25 | 85 | 83 | 86 | 80 | 75 | 84 | 74 | 70 | 77 | 81 | 73 | 88 | 86 | 80 | 91 |
| 50 | 85 | 85 | 84 | 82 | 80 | 84 | 74 | 72 | 75 | 81 | 76 | 86 | 88 | 83 | 91 |
| **75** | **87** | **91** | **84** | **82** | **82** | **82** | **72** | **72** | **72** | **81** | **78** | **84** | **86** | **83** | **88** |
| 100 | 86 | 87 | 84 | 80 | 77 | 82 | 74 | 72 | 75 | 81 | 76 | 86 | 86 | 83 | 88 |

#### 5.5.2.7 FD-Inspired Method

Contrary to all other methods presented in this chapter, the FD-inspired method does not rely on the border distance function (see section 5.4.10). Instead, the FD method calculates an irregularity index (Fdindex) from the division of the lesion perimeter by the perimeter of the fitted shape. We present the results using the fitted ellipse only since the fitted Gaussian did not perform satisfactorily (see section 5.4.10). We will follow the same procedure as in the previously presented experiments by using the same number of interpolated border points using both sampling methods. The experimental results include classification based on single features using our iterative classifier and a set of features using the SVM classifier.

**Single FD Feature Classification using Iterative Classifier:**

Since the FDindex is a single value, it can be used for classification using a newly implemented simple iterative classifier described previously in section (5.5.1.2). The **NoPnts** column in the results tables shows the number of points picked from the lesion border for extracting the FD feature, while **Thresh** column represents the FDindex threshold used by the iterative classifier to split the input images into regular and irregular classes. The thresholds obtained from the internal dataset are then used to test the external dataset.

- **Equi-angdisp Sampling Method**

Table 5.22 shows the experimental results using the Equi-angdisp sampling method. The results show maximum accuracy and the smallest gap between sensitivity and specificity of 86% and 0%, respectively. The performance on external and agreed external cases is still high compared to the previous methods achieving 85% and 90%, respectively. The method outperforms all previously presented methods for internal testing and has the same level of accuracy in external testing as the best one of them.

Table 5.22: Results of FD-inspired method using a different number of border points. (Equi-angdisp sampling).

| **FD-inspired** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | | **External Testing** | | | | | | | | | | | |
| | | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Thresh | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 0.9821 | 65 | 64 | 65 | 64 | 57 | 70 | 60 | 52 | 65 | 61 | 53 | 69 | 66 | 57 | 72 |
| 15 | 1.0025 | 75 | 77 | 73 | 68 | 68 | 68 | 66 | 68 | 65 | 71 | 69 | 73 | 74 | 70 | 77 |
| 25 | 1.0152 | 81 | 81 | 82 | 74 | 68 | 79 | 72 | 68 | 75 | 75 | 67 | 82 | 81 | 70 | 88 |
| 30 | 1.0193 | 82 | 82 | 82 | 78 | 73 | 82 | 74 | 70 | 77 | 79 | 71 | 86 | 85 | 77 | 91 |
| 45 | 1.0255 | 84 | 85 | 84 | 81 | 80 | 82 | 75 | 75 | 75 | 82 | 78 | 86 | 88 | 83 | 91 |
| 90 | 1.0306 | 86 | 86 | 86 | 80 | 80 | 80 | 76 | 78 | 75 | 83 | 80 | 86 | 88 | 83 | 91 |
| 135 | 1.0316 | 86 | 85 | 86 | 81 | 82 | 80 | 77 | 80 | 75 | 84 | 82 | 86 | 89 | 87 | 91 |
| **190** | **1.0321** | **86** | **86** | **86** | **82** | **82** | **82** | **78** | **80** | **77** | **85** | **82** | **88** | **90** | **87** | **93** |

### Equi-arclength Sampling Method

The results shown in Table 5.23 are not far behind the results presented above. However, the method outperforms the method using Equi-angdisp sampling when tested against external agreed cases, achieving a maximum accuracy of 92% among all previously presented methods.

Table 5.23: Results of the FD-inspired method using a different number of border points. (Equi-arclength sampling).

| **FD** SVM-(gamma='scale', kernel='rbf') **Internal Testing** | | | | | **FD External Testing** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **One Doctor** | | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoPnts | Thresh | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 8 | 0.9734 | 64 | 61 | 67 | 64 | 55 | 71 | 60 | 50 | 67 | 61 | 51 | 71 | 66 | 53 | 74 |
| 16 | 1.0057 | 81 | 82 | 80 | 76 | 73 | 79 | 72 | 70 | 73 | 75 | 69 | 80 | 82 | 77 | 86 |
| **32** | **1.0212** | **85** | **85** | **84** | **81** | **84** | **79** | **77** | **82** | **73** | **84** | **84** | **84** | **89** | **87** | **91** |
| 64 | 1.0313 | 83 | 83 | 83 | 82 | 89 | 77 | 78 | 88 | 72 | 85 | 88 | 82 | 92 | 90 | 93 |
| 128 | 1.0368 | 82 | 82 | 81 | 80 | 84 | 77 | 78 | 85 | 73 | 83 | 84 | 82 | 90 | 87 | 93 |
| 256 | 1.0412 | 79 | 80 | 79 | 80 | 84 | 77 | 80 | 88 | 75 | 83 | 84 | 82 | 92 | 90 | 93 |

### Set of FDindex Values as a Feature Vector:

It is interesting to see the effect of using different scales for sampling the interpolated border on the method's performance by using a set of FDindex values calculated using different numbers of sampled points. Therefore, in the following, we present the results of FD inspired method using Equi-angdisp and Equi-arclength sampling methods, respectively. The same range of sampled border point sets of 6 and 8 are used as in the previous experiment for Equi-angdisp and Equi-arclength sampling methods, respectively. The SVM classifier with the same parameters as all previous experiments and evaluation protocol3 is used for the classification.

- **Equi-angdisp Sampling Method**

The experimental results using multi-features and Equi-angdisp sampling, as shown below, are not far behind the ones using single features, achieving 83%, 86%, and 90% on internal, external (doctors C labels), and external agreed cases, respectively.

Table 5.24: Results of FD-inspired method, using a set of FDindexes. (Equi-angdisp sampling method)

| FD SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | | |
| NoFeature | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | |
| 8 | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 | |

- **Equi-arclength Sampling Method**

Contrary to the above results, the method's performance deteriorates significantly on the internal testing set, achieving 79% when the Equi-arclength sampling method is used (see Table 5.25). However, the method outperforms the above results on external agreed cases reaching maximum accuracy of 92%.

Table 5.25: Results of FD-inspired method, using a set of FDindexes. (Equi-arclength sampling method)

| FD SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | | |
| NoFeature | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | |
| 6 | 79 | 80 | 78 | 82 | 91 | 75 | 78 | 90 | 70 | 85 | 90 | 80 | 92 | 93 | 91 | |

The experiments show the high performance of the simple FD-inspired method. The single FD feature's performance was higher than the combined set of FD features. However, their performances on the external testing set were identical, reaching maximum accuracy of 92% on the external agreed cases with the same gap between sensitivity and specificity of 2%. There were only significant performance differences when the two border point sampling methods were used on the internal testing. The method's performance on the internal dataset outperforms all previously presented methods in regard to the gap between sensitivity and specificity of 0% and an accuracy of 86%, which is slightly lower than the TDA-based method.

#### 5.5.2.8 Fast Fourier Transformations FFT

Since the borderline irregularity manifests as fluctuations or frequencies in the distances function curve, it is logical to conduct irregularity analyses in the frequency domain. Therefore, we transform the 1D distances function into the frequency domain using FFT and assess the frequency spectrum for irregularity recognition (see section 5.4.9.1). Any previously mentioned methods for determining the distance function can be used. However, we present the FFT-based method results using only the distances from the centroid and convex hull to reduce the scope of the experiments and for consistency with TDA based method.

Inspired by the work done on the TDA, multiple thresholds are used to count the number of the FFT spectrum spikes for each image (see section 5.4.9.1). The number of spikes is calculated by thresholding the amplitude of the spectrum spikes. For different thresholds, we get a different number of spikes for

each of the lesions. Then, we use the individual and set of spike counts as a discrimination feature for classification using the simple iterative classifier described in section 5.5.1.2 and already used in the FD-inspired method and SVM classifier.

**Single FFT spectrum Features:**

Table 5.26 below shows the method's performance using individual features extracted using one value at a time from the range of 10 amplitude thresholds (0.02 ➔ 0.2). The thresholds obtained from the internal dataset are then used to split the external dataset and to calculate the accuracy, specificity, and sensitivity. The Equi-angdisp sampling method is used to select 180 points from the interpolated border. Then, the border point distances to a centroid form the distance function and are transformed into the FFT spectrum.

The **SpikeThresh** is the frequency spike threshold, and **Thresh** is the ideal feature threshold used by the iterative classifier to split the data.

Table 5.26: Results of FFT-based method using 180 border points. (Equi-angdisp sampling)

| **FFT (Centroid)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | | **External Testing** | | | | | | | | | | | |
| | | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| SpikeThresh | Thresh | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 0.02 | 10.012 | 69 | 56 | 80 | 65 | 45 | 80 | 65 | 45 | 78 | 64 | 45 | 82 | 68 | 47 | 84 |
| 0.04 | 7.0261 | 70 | 61 | 78 | 70 | 61 | 77 | 64 | 55 | 70 | 65 | 55 | 75 | 71 | 60 | 79 |
| 0.06 | 5.008 | 73 | 61 | 83 | 70 | 57 | 80 | 66 | 52 | 75 | 67 | 53 | 80 | 73 | 57 | 84 |
| 0.08 | 4.006 | 73 | 60 | 85 | 72 | 61 | 80 | 68 | 57 | 75 | 69 | 57 | 80 | 75 | 60 | 86 |
| 0.10 | 3.02 | 73 | 55 | 88 | 68 | 48 | 84 | 66 | 45 | 80 | 67 | 47 | 86 | 71 | 43 | 91 |
| **0.12** | **3.02** | **75** | **65** | **83** | **72** | **61** | **80** | **68** | **57** | **75** | **71** | **59** | **82** | **75** | **60** | **86** |
| 0.14 | 3.006 | 73 | 69 | 77 | 68 | 66 | 70 | 64 | 62 | 65 | 67 | 63 | 71 | 70 | 67 | 72 |
| 0.16 | 3.006 | 73 | 78 | 69 | 71 | 75 | 68 | 67 | 72 | 63 | 70 | 71 | 69 | 74 | 80 | 70 |

The method's performance is lower than all other previously presented methods. Using the Equi-arclength sampling method gave similar results and is not presented here.

**Multiple FFT spectrum Features:**

A range of amplitude thresholds (0.02 – 0.2) is used to extract ten spike counts and combine them into one feature vector for input to the SVM classifier. Thus, the search for the best threshold is made redundant, and the experiment is faster because we need to run the experiment one time instead of 10 times. In the following, we present the results of the method using distance functions based on the convex hull and Gaussian, and Equi-angdisp is used to sample 180 border points.

- **Distances from Convex Hull using Equi-angdisp Sampling**

As seen in Table 5.27, the method's performance on internal testing is not better than other best-performing methods; however, its performance on external testing using doctors C labelling reaches the same maximum accuracy of 86% as the best-performing method of FD-inspired.

Table 5.27: Results of FFT-based method, using 180 border points to build distances function from Convex hull. (Equi-angdisp sampling method)

| **FFT (Convex)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoFeature | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |

- **Distances from Gaussian using Equi-angdisp Sampling**

The same above experiment is repeated using the Gaussian shape as a reference for measuring the border distances (see Table 5.28). The performance of the method is deteriorated compared to the convex hull method.

Table 5.28: Results of FFT-based method, using 180 border points to build distance function from Gaussian shape. (Equi-angdisp sampling)

| **FFT (Gaussian, Sigma=6)** SVM-(gamma='scale', kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| NoFeature | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 10 | 73 | 63 | 81 | 73 | 64 | 80 | 67 | 57 | 73 | 72 | 61 | 82 | 77 | 63 | 86 |

The overall results show that assessing the distance function in the frequency domain using FFT transformations gives reasonable performances. Combining ten features extracted from the frequency spectrum using an SVM classifier performs significantly better than the single features using iterative classifiers. Similar to the previous methods using the convex hull as a reference shape yields better performances than the Gaussian shape reaching 78% and 86% accuracy and a 3% and 0% gap between the sensitivity and specificity on internal and external testing sets, respectively. Using more reference shapes and a different number of points from the borderline to build the distance function for FFT analysis might improve the performance, which is interesting to investigate in the future.

## 5.6 Result Analyses

The result of all border distances and FD-based methods are compared in the following, and the best-performing one is chosen by evaluating its overall accuracy and the difference between its specificity and sensitivity on the internal and external testing sets and external agreed cases. To simplify the comparisons, we put the best-performing model from each proposed method in a table for comparison. Due to the high number of methods, we use the method name abbreviations shown in Table 5.29 below in the comparisons.

Table 5.29: Method abbreviations

| Abbreviation | Description |
|---|---|
| **Dist** | Distance based method |
| **Cent** | Centroid |
| **Ellip** | Fitted Ellipse |
| **Gaus** | Fitted Gaussian |
| **Conv** | Convex Hull |
| **ConvEllip** | Fitted Ellipse from Convex Hull |
| **Ftt** | FFT |
| **TDA** | TDA |
| **FD** | Fractal Dimensions |
| **Ang** | Equi-angdisp sampling method |
| **Arc** | Equi-arclength sampling method |

Table 5.30 below describes three colours to rank the compared methods according to their performances as first, second and third best.

Table 5.30: Different colours to show the best-performing methods.

| Color | Performance |
|---|---|
| | Best |
| | Second Best |
| | Third Best |

Table 5.31 shows that the FD-inspired method using the iterative classifier (red coloured) outperforms all other methods when tested internally and for the first two doctors' class labels when tested externally. The TDA-based method performs second best on internal testing and best when doctor A's class label is used on external testing. The method based on the fitted ellipse from the convex hull still outperforms all other methods when tested externally using doctor C's ground truth, reaching 88% and 0% accuracy and a gap between sensitivity and specificity, respectively. The third best-performing method is still FD-inspired, using a set of features as input to the SVM classifier. All other methods are not far behind, with fitted ellipse and FFT using distances from the centroid at the end of the performance rank.

Table 5.31: Methods performance comparison when tested on the internal and external sets.

| Method | Classifier | Internal Testing | | | External Testing | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| **Dist** (**Cent**.) Ang. | Svm (rbf) | 82 | 83 | 81 | 75 | 70 | 79 | 73 | 70 | 75 | 76 | 69 | 82 |
| **Dist** (**Ellip**.) Ang. | Svm (rbf) | 78 | 83 | 73 | 74 | 77 | 71 | 72 | 78 | 68 | 77 | 78 | 76 |
| **Dist** (**Ellip**.) Arc. | Svm (rbf) | 73 | 78 | 69 | 70 | 80 | 62 | 70 | 82 | 62 | 79 | 86 | 73 |
| **Dist** (**ConvEllip**.) Ang. | Svm (rbf) | 83 | 80 | 86 | 76 | 77 | 75 | 74 | 78 | 72 | 83 | 82 | 84 |
| **Dist** (**ConvEllip**.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 |
| **Dist** (**Gaus**.) Arc. | Svm (rbf) | 80 | 80 | 80 | 73 | 68 | 77 | 71 | 68 | 73 | 76 | 69 | 82 |
| **Dist** (**Conv**.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 |
| **Dist** (**Conv**.) Arc. | Svm (rbf) | 82 | 89 | 77 | 76 | 84 | 70 | 76 | 88 | 68 | 85 | 90 | 80 |
| **TDA, Dist** (**Conv**.) Arc. | Svm (rbf) | 87 | 91 | 84 | 82 | 82 | 82 | 72 | 72 | 72 | 81 | 78 | 84 |
| **FD** Ang | Iterative | 86 | 86 | 86 | 82 | 82 | 82 | 78 | 80 | 77 | 85 | 82 | 88 |
| **FD** Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 |
| **FFT, Dist** (**Cent**.) Ang. | Iterative | 75 | 65 | 83 | 72 | 61 | 80 | 68 | 57 | 75 | 71 | 59 | 82 |
| **FFT, Dist** (**Conv**.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 |

In Table 5.32, we present the models performing the best on the external agreed doctor's class labelling. In contrast to the above comparison, we select the best models performing best on external agreed cases regardless of the model's performance on internal or external testing sets when individual doctors' class labels are used. Both methods of FD-inspired based on a set of features and fitted ellipse from the convex hull perform the best, reaching 92% and 2% accuracy and a gap between sensitivity and specificity, respectively. The FD-inspired method using a single feature with the iterative classifier still performs second best, achieving the same total accuracy but with a slightly higher gap between sensitivity and specificity. The distance from the fitted ellipse from the convex hull using Equi-angdisp sampling performs third best on the performance ranking.

Table 5.32: Methods performance comparison when tested on the external agreed cases between the three doctors.

| Method | Classifier | Internal Testing | | | External Testing Agreed Cases | | |
|---|---|---|---|---|---|---|---|
| | | Acc | Reg | Irreg | Acc | Reg | Irreg |
| **Dist** (**Cent**.) Arc. | Svm (rbf) | 74 | 78 | 70 | 89 | 93 | 86 |
| **Dist** (**Ellip**.) Arc. | Svm (rbf) | 72 | 78 | 67 | 81 | 80 | 81 |
| **Dist** (**ConvEllip**.) Ang. | Svm (rbf) | 78 | 76 | 80 | 90 | 90 | 91 |
| **Dist** (**ConvEllip**.) Arc. | Svm (rbf) | 78 | 76 | 80 | 92 | 93 | 91 |
| **Dist** (**Gaus**.) Ang. | Svm (rbf) | 76 | 74 | 78 | 86 | 77 | 93 |
| **Dist** (**Conv**.) Arc. | Svm (rbf) | 84 | 91 | 78 | 88 | 90 | 86 |
| **TDA, Dist** (**Conv**.) Ang. | Svm (rbf) | 81 | 85 | 78 | 89 | 87 | 91 |
| **FD** Arc | Iterative | 83 | 83 | 83 | 92 | 90 | 93 |
| **FD** Arc | Svm (rbf) | 79 | 80 | 78 | 92 | 93 | 91 |
| **FFT, Dist** (**Conv**.) Ang. | Svm (rbf) | 78 | 80 | 77 | 85 | 80 | 88 |

Overall, the FD-inspired method performs the best, followed by TDA and distances method based on either the convex hull or fitted ellipse from the convex hull. All methods show high performances; however, parameters such as the number of sampled points or the threshold values in the case of the FFT and TDA-based methods still need to be selected empirically, and it is not easy to find a set of parameters fitting for all testing scenarios.

### 5.6.1 FD using Box-Counting Method

The FD based on the box-counting method (see section 3.4.4) for border irregularity assessment did not give satisfactory results, perhaps due to the fact that it only assesses the sampled points on the interpolated borderline but not the pixel intensities around the border. Several recent works reported in the literature [156], [163] where the fractal dimensions are calculated from the geometrical location of the border points (x and y coordinates) as well as the pixel intensities in a box around the border points. This is an investigation worth attempting since the pixel intensities around the borderline include valuable information about the actual course of the border of the lesions and, consequently, might discriminate the border irregularity.

## 5.7 Concluding Remarks

In this chapter, we developed and tested the performance of a variety of AI schemes to determine the irregularity of thyroid lesions border in US scan images. Two different approaches were developed: (1) methods based on vectorizing distance functions between interpolated border curves and known regular curves associated with ROI-marked border points and (2) FD-inspired parameters. Our results demonstrated the viability of determining the irregularity of nodule border from US scan images using a small set of ROI points instead of manual/automatic segmentation. Several reference shapes, such as fitted ellipse, fitted Gaussian, convex hull, and fitted ellipse from the convex hull, is used for border distance measurements, and their performances are compared. The distance function is either directly used as a feature vector, transferred into the frequency domain, or analysed by TDA for extracting the feature vector. FD is the only method not based on border distances, which calculates an irregularity index for classification use. Furthermore, one evaluation protocol is developed, delivering one best model from 100 trained ones for internal and external testing. A simple iterative classifier is developed for classifying methods such as FD and FFT based on single irregularity features. All methods are evaluated on internal and external testing sets as well as on external sub-testing sets with agreed labels between three radiologists. All methods achieved average accuracies above 73%, with FD inspired method using the iterative classifier achieving the highest accuracy of 86% on internal testing and the method based on the convex hull achieving maximum accuracy of 88% on external testing with doctors Cs ground truth. The maximum accuracy on external agreed cases was 92% achieved by both FD-inspired and ellipse from convex hull-based methods.

This chapter included no information on Textures/Pixel intensities from the region bounding the interpolated lesion border. The next chapter is devoted to the task of exploiting texture information in the bounding region.

# Chapter 6: Texture Analysis for Thyroid Nodule Border Irregularity

In the previous chapter, we developed and tested several AI algorithms for irregularity recognition of thyroid nodule borders using the bicubic interpolation of a finite set of ROI points marked by the clinician rather than segmenting the nodule. We successfully found several high-performing schemes, but the observation that, for some cases, the poor visibility of the border region may partially explain the misclassification cases in terms of our interpolation approach for nodule border estimation. In this chapter, we investigate the use of texture analysis of the tissue images in the vicinity of the interpolated border to design and test the performance of a few such schemes for border irregularity recognition. In particular, we use the previously investigated LBP, HOG, and HOL textures for detecting abnormalities in glass façades and concrete surfaces, but to be extracted from a ribbon around the interpolated lesion border of reasonable margin. We also investigated CNN architectures based on transfer learning for lesion border irregularity recognition, but in opposite to our first case study of crack recognition, transfer learning by fine-tuning approach is used. Figure 6.1 below shows an overview of our proposed methods and their sub-sections described in this chapter.

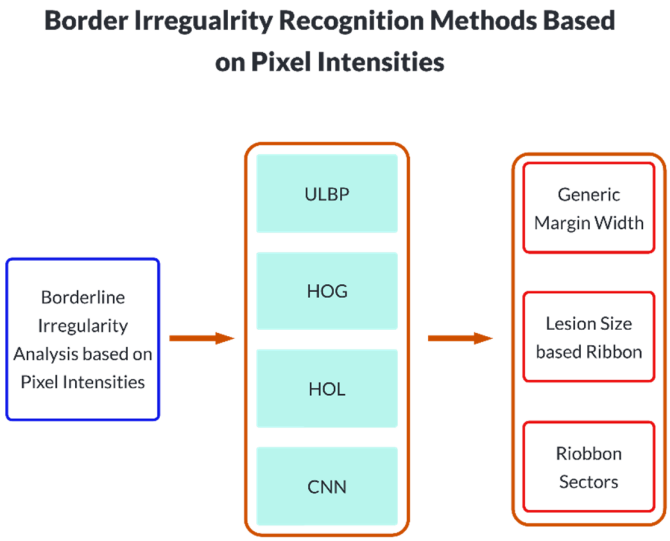



Figure 6.1: Summary of the proposed methods based on texture analysis.

## 6.1 Introduction

The border estimation from the ROI points presented in the previous chapter did not rely on thyroid nodule-related image data (pixel intensities) but instead relied on the positions of a finite set of points in a 2-dimensional rectangle. Notwithstanding the success of the irregularity schemes developed using this approach, a natural question arising is whether image texture information in the surrounding regions

of the lesion border can be used to determine border irregularity. Furthermore, using such information may compensate for the possibility of the interpolated border not providing a good estimate of the actual border.

Therefore, in this chapter, we shall investigate border irregularity using texture features extracted from the immediate surrounding regions of the interpolated lesion border. These investigations start by building a ribbon around the interpolated lesion border and selecting texture feature vectors from the ribbon. Rather than investigating all sorts of texture features, we shall be guided by the success achieved when we investigated the abnormality of cracks in glass façades and concrete blocks and shall confine our investigations to the use of Histogram of Linearity (HOL), HOG, and LBP based texture features. Both LBP and HOG-based methods are used for border irregularity recognition to extract discriminating texture features from border ribbons of different widths. In contrast, the HOL method is based on edges extracted from the regions surrounding the borderline using the ribbon of different widths.

In the last chapter, we observed that the ROI interpolated lesion border of extremely shaped nodules or small lesions might result in difficult-to-analyse borders, e.g., the line of view from the centroid to some ROI points passes through other border points on their way. Therefore, border ribbon construction must avoid creating self-intersecting ribbons to simplify the texture feature extraction process with no duplications. Besides using the whole ribbon, the ribbon can be split into several sectors before feature extraction. We will additionally extract these texture features from the entire lesion bounding box in the discussion section to conclude our investigations, and we will compare the outcomes with those from the ribbons.

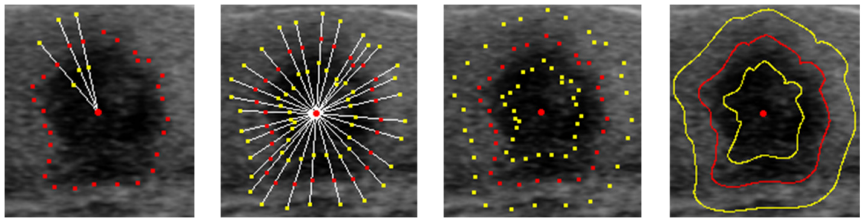
Furthermore, we shall evaluate the performance of some common CNN techniques for border irregularity assessments of cancer lesions. Since our datasets are small for training a CNN model from scratch, our attempted CNN models are based on transfer learning.

## 6.2 Data Preparation: Constructing Lesion Border Ribbon

As lesion boundary regularity/irregularity can be seen as a spatial relationship between the values of the pixels on the lesion border and the values of the pixels nearby, the logical thought will be to examine texture patterns or repeated pixel intensity change patterns close to the border of the lesion marked by the ROI points. Therefore, a band (or a ribbon) is built around the ROI points. Since no segmentations of the dataset are provided, and the lesion borderline is not precisely known; therefore, ribbons of different widths are attempted to find the best-performing one. However, the highly irregular border may result in self-intersecting ribbons rendering many of the scales unusable. Two methods based on radial distances and morphological operations (e.g., dilation and erosion) have been attempted to build the ribbon, which will be described in the following sections.

### 6.2.1 Border Ribbon Construction using Radial Distances:

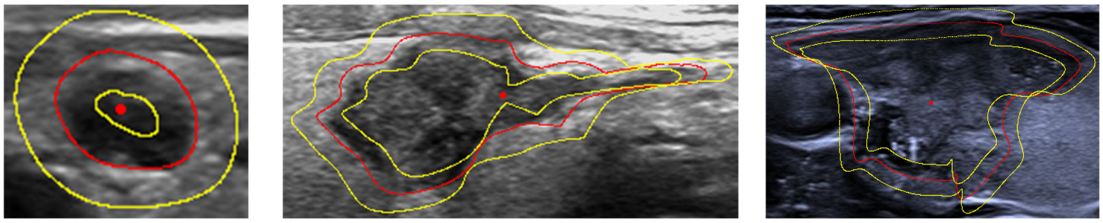
A simple radial line-based method is used for building a ribbon around the border. First, lines are drawn from the centroid of the nodule to the individual original ROI points (see Figure 6.2 a, red dots). Then, from each of the ROI points along the line (white line in Figures 6.2 a and b), a certain distance (in pixels) depending on the required ribbon width is measured inwards and outwards to mark the points on the inner and the outer margin (yellow points). Finally, the margin points are interpolated using any interpolation methods (in our work, we use cubic spline) to build the ribbon for use in methods such as ULBP, HOG, HOL, and CNN (see Figure 6.2 d).



(a) (b) (c) (d)

Figure 6.2: Ribbon using radial lines (a) radial lines from the centroid to the original ROI points (red dots) (b) inner and outer ribbon margin (yellow dots) (c) marked inner and outer margin (d) interpolated ribbon margin (yellow curves).

The radial distances-based ribbon works well for most of the nodules across our datasets, mostly regular cases (see Figure 6.3 a). However, uneven ribbon widths around the boundary can be observed in some extreme cases of lesion shapes (see Figures 6.3 b and c). Further, we can also see from the figures that the two inner and outer margins are crossing in some locations where there is a sharp turning in the boundary. We use a new method based on the morphological operations described in the next section to overcome these shortcomings.



(a) Regular (b) Irregular (c) Irregular

Figure 6.3: Ribbons for various regular and irregular nodule shapes.

### 6.2.2 Ribbon based on Morphological Dilation and Erosion

This method uses established morphological erosion and dilation operations to be applied to the image region centred at the sampled border points to build the ribbon. The choice of these operations is meant to avoid having small holes, overlapping, and or isolated individual outlier pixels. Figure 6.4 shows the process of building the ribbon. First, an erosion operation is conducted starting from the interpolated borderline using a disk of a certain thickness. This way, the boundary of the nodule will uniformly

shrink on all sides independent of any elongations or sharp border turnings, producing an erosion mask (see Figure c). Similarly, a dilation operation is started from the interpolated borderline using a disk of the same size producing a dilated mask (see Figure b). The boundary points of the eroded mask edge will mark the inner margin, while the dilated mask edge marks the outer margin of the ribbon (see inner and outer yellow curve Figure d). The inner and outer margins are used to mark the ribbon region and to crop the ribbon for the input to CNN models, where the disc size determines the ribbon width.

(a) (b) (c) (d)

Figure 6.4: Building masks using morphological erosion and dilations (a) original ROI mask (b) dilated mask (c) eroded mask (d) ribbon marked by two yellow curves.

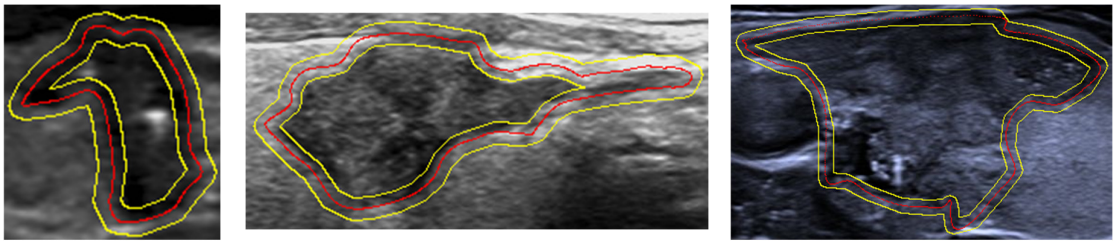


Figure 6.5: Some irregular cases with extreme shapes.

Unlike the radial distances approach, with few exceptions, morphology-based border ribbon construction results in ribbons of nearly constant thickness. The images in Figure 6.5 demonstrate this observation compared to those obtained in Figure 6.4 by the radial distances approach. Narrowly elongated border regions may be the exception but can be treated using thinner widths.

## 6.3 Texture Analysis for Nodule Border Irregularity Recognition

Several of our methods are presented in the following sections. All the methods, including the CNN models, are based on the ribbon described previously to restrict the texture feature extraction around the lesion borderline for irregularity recognition. Figure 6.6 below shows an overview of the main steps of our handcrafted methods for border irregularity recognition of thyroid nodules. The steps include; (1) building the ribbon from the ROI points, (2) dividing the ribbon into sectors of equal radial distances, (3) extracting features from sectors, (4) building a histogram by concatenating the sector histograms, and (4) feeding the histogram to a classifier.

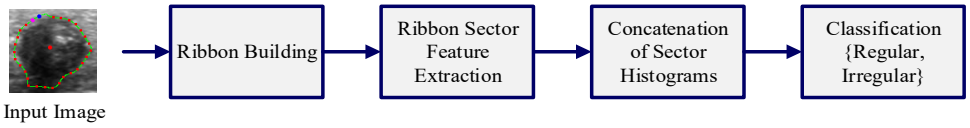



Figure 6.6: Overview of the texture analysis-based methods for irregularity recognition of thyroid nodule border.

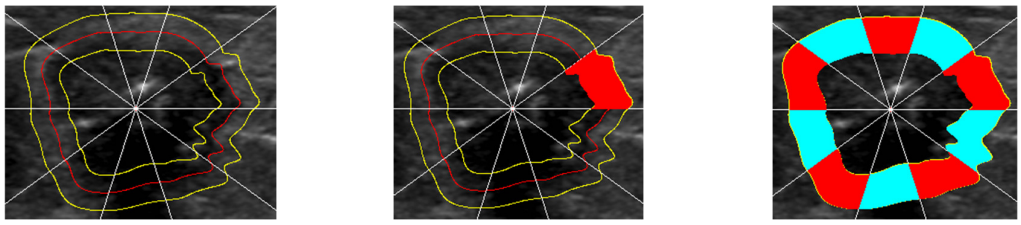
### 6.3.1 LBP Texture Feature

In section 3.4.2, we described the LBP texture features and used them in section 4.4.4 for our first case study of glass and concrete crack classification (see [36], [98]). The LBP is extensively used in various application domains, including medical image analysis showing high performances [19], [100], [174]. In US images, the nodule border is visually represented in texture abnormality or pixel intensity variations around the nodule boundary. Hence irregularity of the border can be considered a texture analysis task and can be done using texture feature-based methods such as LBP and HOG. Hence, we use LBP-based texture analysis in our nodule border irregularity recognition for thyroid cancer. The basic approach here is to extract LBP features (codes) from the region of interest pixels and then build a histogram from various LBP codes and use it as input to a chosen classifier. The bounding box and the ribbon around the border could be used as the region of interest, although using the ribbon around the borderline is more realistic since we analyse border irregularity. Similar to the crack recognition methods (see section 4.4.8), the whole region (ribbon region in our case)- or partition-based LBP can be used. Partition-based LBP method proved effective for glass and concrete crack recognition; hence it is interesting to see its effectiveness in border irregularity analysis of cancer nodules. Again, only the ULBP (all groups or single $G_i$ groups) are used instead of all the LBP codes to reduce feature vector dimensionality and reducing computational costs. Also, interesting to exclude the two codes 255 and 0, representing light and dark 3x3 patches from the 58 ULBP codes for better performance. In the experimental work presented in section 6.4.1, we extract the chosen ULBP feature vectors in the following Regions of Interest:

- The entire nodule tissue image is bounded by the interpolated border
- The constructed interpolated border ribbon
- The interpolated border ribbon partitioned into sectors

Partitioning the input image into equal blocks before extracting the ULBP histogram for the glass and concrete crack recognition improved performance by about 5%. Therefore, the same partitioning approach for irregularity recognition of cancer lesion border based on the ULBP (or other texture feature) method is worth attempting. This can be done by dividing the whole ribbon around the lesion into sectors. Figures 6.7 a-c show a simple method based on lines of equal radial distances to divide the ribbon into sectors (see white lines). The lines are drawn from the centroid (or centre of a fitted ellipse) at equal radial distances crossing both the inner and outer margins of the ribbon (see yellow curves).

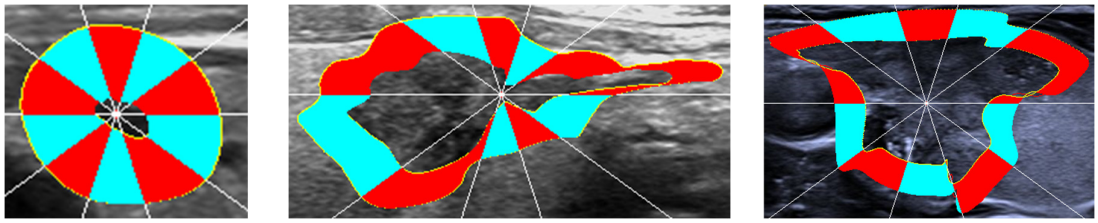
Then, the intersection points are used to draw the sectors. The colours red and blue mark different sector regions.



(a) (b) (c)

Figure 6.7: Building sector-wise ribbon partitions.

Figure 6.8 below shows more cases of regular and irregular lesions partitioned, for instance, into ten equal sectors. The white lines are drawn in 360/10=36° radial distances to get ten sectors. Contrary to partitioning the bounding box, this method ends up with sectors of different sizes depending on the nodule's shape. The differences in sector sizes are higher for lesions of more irregular shapes (see Figures b and c) and lower for more regular or elliptical nodule shapes, as in Figure a.



(a) Regular (b) Irregular (c) Irregular

Figure 6.8: Ribbon sectors of regular and irregular lesions.

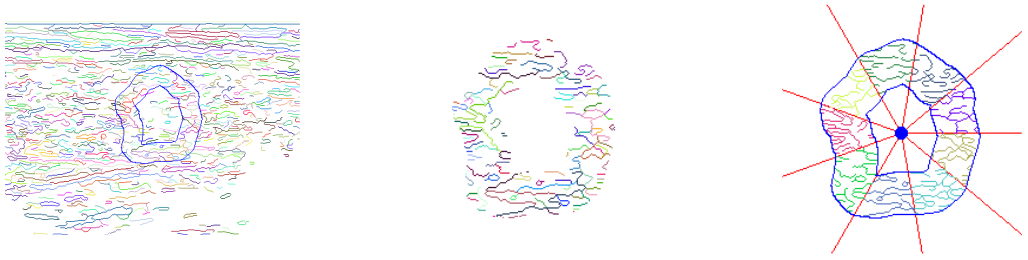
As a feature vector, a histogram of the ULBP codes is extracted from the individual sectors. Each sector histogram is normalized by dividing each bin by the sum of all bins. Then, the sector histograms are concatenated to build, for instance, a feature vector of 10x58 if ten sectors and ULBP(58) are used.

### 6.3.2 HOG Texture Feature

The HOG-based method described in section 3.4.3 was also used effectively in glass and concrete crack recognition (sections 4.4.5) from visual images, where its performance lay slightly behind the ULBP-based method on the glass but superior on the concrete. The HOG feature vector is used for border irregularity recognition post applying the gradient transform on the grayscale image. While LBP is based on local neighbourhood pixel intensity comparison, HOG uses intensity gradients and directionality of the local pixel neighbourhoods. Like the ULBP-based method, we divide the input ribbon image into nine non-overlapping sectors, build a histogram of nine bins for each sector, and concatenate them to form 9x9 bins feature vector before feeding it to a classifier.

### 6.3.3 HOL Texture Feature

The Histogram of Linearity (HOL) method was introduced in chapter 4 as a generalization of the HOG texture feature, whereby first, the special thin edge detection algorithm (ED algorithm [64]) was applied to the input images, and the statistics of the linear fit to the extracted edges was represented by a histogram of certain finite numbers of bins. The HOL was used to recognise glass and concrete cracks (sections 4.4.2) from recorded natural images and was an effective discriminating feature. Here, we propose the use of HOL as a predictor for nodule border irregularity recognition. We shall apply this to the entire nodule ribbon tissue image and border ribbon sector regions. Figure 6.9 below shows the ribbon and the sectors, whereby different colours are used just for visualization of the different ED edge segments. Like the other two previous methods, a histogram of linearity values extracted from each ribbon sector is constructed and normalized. Then, the sector histograms are concatenated to form a feature vector for input to a chosen classifier.



(a) (b) (c)

Figure 6.9: ED edge segments drawn in different colours (a) ribbon margins in blue (b) ED edges cropped using ribbon (c) ribbon sectors.

### 6.3.4 Deep Learning-based Border Irregularity Recognition

Deep Convolutional Networks described in section 2.3 are state-of-the-art techniques for many computer vision tasks, including cancer diagnosis, as described in detail in the two surveys on different types of cancer [175], [176]. Hence, we attempt to use CNN models for border irregularity recognition of thyroid nodules. Since our nodule border dataset is relatively small to train a CNN model from scratch, we train the CNN models using transfer learning, similar to the abnormality recognition in the case study of cracks in building material presented in chapter 4. However, the transfer learning by fine-tuning approach is used in opposition to the transfer learning by feature extraction. In the transfer learning by feature extraction approach (see section 4.4.9), the already trained convolutional layers of the CNN model are used for feature extraction without retraining. The extracted features are then fed to new fully connected layers designed for the new classification task with two classes in the case of cracks. This type of transfer learning was sufficient in the case of glass and concrete since we are dealing with natural images. This method is inadequate for US images due to the weak textural variations in US images. Therefore, the CNN model needs to be retrained using transfer learning by finetuning on the new dataset, starting with weights already trained on the natural images such as ImageNet [37]. The

model learned general image features from the natural images and can learn new specific features unique to the thyroid cancer US images. The same CNN architectures as in the case study of crack recognition, VGG16 [145] and ResNet50 [146] are used for consistency. As described in section 4.4.9, the two architectures come from two different CNN families where the VGG16 uses a sequential approach for building the layers while ResNet50 uses skip connections to allow very deep layers.

Contrary to the crack recognition, the thyroid US lesion borders are cropped before using as input to the CNN models. The cropping is done using a bounding box combined with the ribbon approach (see Figure 6.10). The same ribbon method based on morphological operations is used with different ribbon widths as in the other handcrafted methods, such as ULBP, HOG, and HOL. However, to reduce the scope of the experiments, we used only two ribbon widths of 12 and 18 pixels. In this sense, we first build the ribbon using certain ribbon width. Then, we crop the nodule using a bounding box around the outer margin of the ribbon. Finally, we set all the pixels outside the ribbon to 0 to ensure the CNN model focuses on the pixel intensity variations around the borderline. Notice that the interpolated border line in green and the ROI points in red shown in Figure 6.10 below are just for visualisation and not included in the input image to the CNN.

(a) (b)

Figure 6.10: Ribbon bounding box for CNN border region cropping. Original ROI points red dots, interpolated border greed curve (a) ribbon width 12 pixels (b) ribbon width 18 pixels.

## 6.4 Experimental Results and Evaluations

The following sections present the experimental results using the methods based on pixel intensity analysis (ULBP, HOG, and HOL) and the CNN models. The datasets described in section 5.3 are used to evaluate the proposed methods. The internal dataset DS(395) is used as training and internal testing sets, while the smaller dataset DS(100) is used as external testing. We need to remember that the external dataset is provided with three radiologists' class labelling. We use evaluation protocol3 for most of the proposed handcrafted methods and protocol1 (five-fold cross-validation) for a few of the experiments on ULBP methods and CNN models. We need to remember that protocol3 chooses one model from 100 models trained on randomly selected training/evaluation sets from the internal dataset using the highest accuracy and smallest gap between sensitivity and specificity as selection criteria. We

attempted to use kNN and SVM classifiers; however, the SVM performed better in most of the experiments. Therefore, for consistency, we present only results using an SVM classifier with the same parameters. As in previous experiments, the average accuracy (**Acc**), specificity (**Reg**), and sensitivity (**Irreg**) are recorded in each of the experiments.

We use different ribbon widths from 1 to 24 pixels throughout our experiments; however, we present here only the two best-performing ribbons on internal and external testing for each experiment. For the CNN models, we present the five folds' average accuracy, specificity, and sensitivity. The full results of the range of the ribbon widths and the five-fold cross-validations are attached in Appendix B.

### 6.4.1 ULBP based Method

In the following, we present and compare the results of ULBP-based methods using different approaches. Most of these approaches are applicable to the other handcrafted feature-based methods such as HOG and HOL, but due to the scope of the experiments, they have only been applied to the ULBP-based method. These approaches include ULBP based on the whole ribbon and ribbon sectors, ULBP code groups, training on multiple ribbon widths, and training on different lesion size categories. In addition, we present and compare the results using the ribbon cropping methods based on radial distances and morphological erosions and dilation. Several numbers of ribbon sectors are employed in our tests. However, nine sectors produce superior overall accuracy in most of our experiments, and thus, for consistency and comparability, we utilise nine sectors in all of our evaluations. Finally, different margin widths are attempted to find the best-performing values.

#### 6.4.1.1 ULBP based on the Whole Ribbon

Nodule border ribbons of different widths are constructed around the ROI points using both the radial and morphology methods (see section 6.2). A histogram of the ULBP codes calculated from the pixels inside the ribbon is formed for each thyroid case. The histogram is normalized by dividing each bin by the sum of all bins for input to a classifier. Table 6.1 show the results using two ribbon methods (**Rbn.Method**) based on radial distances (**Radial**) and morphology (**Morph**). The two best-performing ribbon width (margin) results are shown for each of the ribbon methods.

Table 6.1: ULBP method based on the whole ribbon. (see Appendix B.1 Tables 1 & 2)

| **ULBP(58) whole ribbon** (gamma='scale', kernel='poly', degree=9) | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | | **External Testing** | | | | | | | | |
| | | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | |
| Rbn.Method | Margin | Acc | Reg. | Irreg. | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Rdial | 4 | 60 | 50 | 69 | 64 | 57 | 70 | 64 | 57 | 68 | 65 | 57 | 73 |
| Rdial | 18 | 65 | 65 | 66 | 56 | 43 | 66 | 56 | 43 | 65 | 51 | 39 | 63 |
| Morph | 5 | 59 | 54 | 64 | 57 | 50 | 62 | 61 | 55 | 65 | 60 | 53 | 67 |
| Morph | 10 | 65 | 63 | 67 | 57 | 52 | 61 | 61 | 57 | 63 | 54 | 49 | 59 |

The test results above show similar overall achieved accuracies of around 65%, with a small gap between sensitivity and specificity for both radially and morphologically constructed ribbons. However, testing results over the external dataset reveal slight differences. Although both ribbon methods were used in the remaining experiments, only the results based on the morphological ribbons are presented for better fitting, especially in the extreme shape lesions and for consistency.

#### 6.4.1.2 Ribbon Sectors-based ULBP

Table 6.2 presents the results of the sector-based ULBP method using the morphological ribbon cropping method. These results show a significant performance improvement of the ULBP scheme, yielding the highest accuracy of 72% on the internal dataset using two ribbon widths of 20 and 22 pixels. However, ribbon width 22 gives a smaller gap between sensitivity and specificity. This is a significant increase in the method's performance by at least 10% compared with the ULBP based on the whole ribbon (see Table 6.1). The performance improvements could be due to certain discriminative features of the border captured in different locations around the boundary and used to classify regular and irregular lesions. In addition, the improvements can also be due to an increase in feature dimensionality through partitioning. The trained models achieve similar overall accuracies, around 73% (Doctor B ground truth), with a higher gap between sensitivity and specificities when tested on the external testing set. Furthermore, it can also be observed that the sensitivity is significantly higher than the specificity in the case of external testing.

The variation in the performances of the scheme on the external dataset in relation to the different expert assessments of the same set may be related to variation in their experience, but to see how much this variation is to be, we repeated the testing on the subset of the external dataset were the three doctors agreed on the labelling (see Table 6.2). The results in the table demonstrate that the overall accuracy improves to 78%, and sensitivity and specificity increase significantly compared to individual labelling of the ground truths. These results highlight the seriousness of testing the machine learning algorithm's performance for medical image diagnostic tasks in the absence of clinically agreed class labelling among medical experts.

Table 6.2: Ribbon sectors based ULBP using morphological ribbon construction. (see Appendix B.2 Table 3)

| **ULBP(58) ribbon sectors** SVM (C=100, degree=1, gamma=0.1, kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Margin | Acc | Reg | Irreg | Acc | Acc | Reg | Irreg | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 20 | 72 | 63 | 80 | 67 | 55 | 77 | 73 | 62 | 80 | 70 | 57 | 82 | 78 | 67 | 86 |
| 22 | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |

#### 6.4.1.3 ULBP based on Code Groups:

Guided by our work in chapter 4, we experimented with using the single groups of ULBP to test if we could make the ULBP scheme slimmer while maintaining or improving their performance. The experiment revealed that only the G3 group (i.e., sets of ULBP codes that consists of 3 consecutive 1's only) could provide a chance for a slimmer scheme. Not that the histograms of each ULBP group consist of only eight bins compared to 58 bins representing the whole ULBP code range.

The performance of the slimmer G3 scheme on the internal and external datasets is presented in Table 6.3 below. Note that the performance of the G3 scheme is similar to the ULBP (58) codes achieving a maximum overall accuracy of 72% on both internal and external testing (doctor C ground truth). Thus the G3 ULBP scheme presents much slimmer feature vectors with the same performance as the total ULBP codes.

Table 6.3: G3 tested on the external testing set. (see Appendix B.3 Table 4)

| **ULBP(G3) ribbon sectors** SVM (C=100, degree=1, gamma=0.1, kernel='rbf') | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | |
| Margin | ACC | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 8 | 69 | 74 | 64 | 65 | 59 | 70 | 71 | 68 | 73 | 72 | 65 | 78 |
| 11 | 72 | 72 | 72 | 64 | 52 | 73 | 60 | 48 | 68 | 61 | 49 | 73 |

#### 6.4.1.4 Model Training using Multiple Generic Margin Widths

The above experiments show that certain ribbon widths may give the best performance on one dataset while giving a lower performance on another. An interesting question that arises here is which generic ribbon width should be used in the training and external model testing. The external dataset has different nature, and therefore it may work better with varying widths of the ribbon. One solution for this issue could be to train a model using several copies of the same training image but with different ribbon widths. This can be thought of as an augmentation of the training set. Thus, the model is more generalized in terms of the ribbon width and performs better on the external dataset using any width from a range of ribbon widths.

Using protocol3 in this approach requires using copy samples only in the training sets, not in evaluation sets. This means we only create copies of the training sets in each of the 100 random splits (80% training/ 20% evaluation). Not to mention that the 30% testing set separated for internal testing at the beginning of the protocol is also not duplicated. Table 6.4 below shows two models (model1 and model2) trained using two sets of five ribbon widths (**Training Margins**). The gap between the widths in the first set is 4 pixels, while in the second set is 1 pixel. These two-ribbon width sets and their width gaps are empirically chosen to find the best values. The range of margin widths is only used for the 100 random training sets. In contrast, different fixed ribbon widths of 22 and 18 pixels are used for both

model1, and model2 evaluation sets (**Int.Eval.Margin)**, respectively and a fixed ribbon width of 18 is used for both generic models' internal testing (**Int.Test.Margin)**,. Table 6.4 below presents the two models' performance on the internal testing set, while Tables 6.5 and 6.6 show the performance of the two models when tested on the unseen external dataset with three class labels and agreed labels.

Table 6.4: Two models trained on generic ribbon widths. Testing results on the internal dataset.

| **ULBP (58) /** Generic Ribbon Size<br>SVM (C=100, degree=1, gamma=1, kernel='rbf')<br>**Internal Testing** | | | | | | |
|---|---|---|---|---|---|---|
| **Model** | **Training Margins** | **Int.Eval.Margin** | **Int.Test.Margin** | **Acc** | **Reg** | **Irreg** |
| 1 | 10,14,18,22,26 | 22 | 18 | 65 | 69 | 62 |
| 2 | 16,17,18,19,20 | 18 | 18 | 68 | 70 | 66 |

**Testing Model 1 on the External Dataset:**

Table 6.5 below shows significant improvement in the method's performance on the external dataset on all three class labels, with the highest accuracy of 77% on doctor C. The accuracy improvement was about 6% compared with the method's performance when fixed ribbon widths were used to train the model. The model's performance reached 82% on the external agreed cases between the three class labels, with a relatively small gap between sensitivity and specificity.

Table 6.5: Model1 testing on the external dataset. (see Appendix B.4 Table 5)

| **External Testing**<br>**Model 1** | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Margin | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 16 | 74 | 68 | 79 | 74 | 70 | 77 | 77 | 69 | 84 | 82 | 73 | 88 |
| 17 | 74 | 70 | 77 | 74 | 72 | 75 | 77 | 71 | 82 | 82 | 73 | 88 |

**Testing Model 2 on the External Dataset:**

The model2's performance on the external dataset shown in Table 6.6 achieves the best accuracy of 72% on doctor C's ground truth. This performance is lower than model1 by 5% but similar to the models when the fixed ribbon was employed. The best model performance on agreed cases is 75%, which is again lower, about 7% from the model1. The results show that choosing a range of ribbon widths with a wider gap (model1) between the widths performs better on the unseen data than narrow ones (model2).

Table 6.6: Model2 testing on the external dataset. (See Appendix B.4 Table 6)

| **External Testing**<br>**Model 2** | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Margin | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 17 | 69 | 64 | 73 | 69 | 65 | 72 | 72 | 65 | 78 | 75 | 70 | 79 |
| 18 | 68 | 66 | 70 | 66 | 65 | 67 | 69 | 65 | 73 | 73 | 70 | 74 |

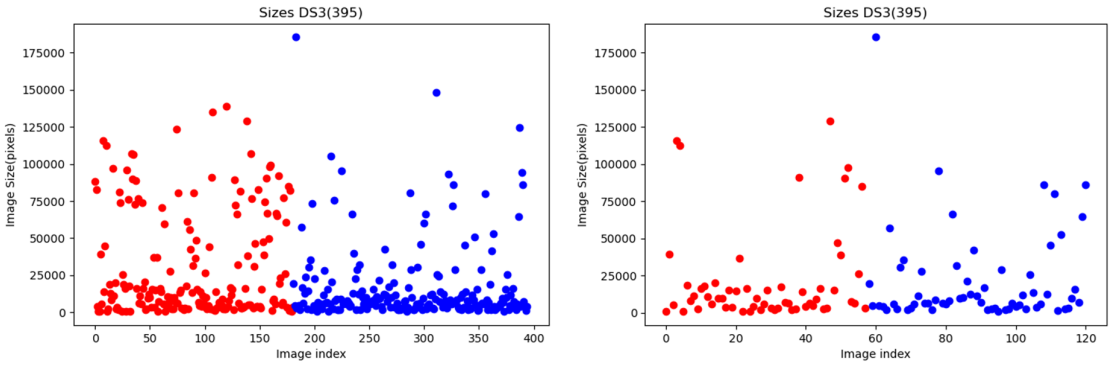
#### 6.4.1.5 Analysis of ULBP Ribbon Width vs Lesion Size

There is a high variation in lesion sizes across our datasets, starting with 538 pixels for the smallest lesion to 185340 pixels for the largest one. The lesion size is calculated by counting all pixels inside the nodule boundary (see the yellow region in Figure 6.11 b). Since the border irregularity analysis must involve the texture or pixel intensity variations around the border, the width of the ribbon used for texture analysis must cover the boundary regions. However, the true amount of boundary regions is unknown due to the absence of accurate border segmentation. It is natural to assume that small lesions will have smaller boundary regions and large lesions will have larger boundary regions. Therefore, it makes sense to calculate the ribbon width based on the lesion size for boundary texture analysis.

(a) (b)

Figure 6.11: Calculating lesion sizes for training models based on size categories of the lesions.

We attempted to analyze the lesion size distribution across our datasets. Figure 6.12 show the size distribution of the entire internal dataset and the missed classified cases using the ribbon sector-based ULBP method. Most of the nodules have smaller sizes, under 25000 pixels, in both regular (red dots) and irregular (blue dots) classes (see Figure a). When it comes to the miss classification cases, again, most of the nodules come under the size of around 25000, and both classes show similar distributions (see Figure b). The plots indicate less dependence of the sizes on the nodule class, although the regular cases tend to be more spread into bigger sizes, as can be seen in Figure a.




(a) Entire dataset. (b) Missed cases, ribbon 10.

Figure 6.12: Lesion size distribution among all cases and missed classified cases using the ULBP method. Regular (red dots), Irregular (blue dots).

#### 6.4.1.6 Model training depending on different Lesion Size categories

As the first attempt to investigate the effect of the lesion size on the ribbon width, we divide our internal dataset of 395 images into three size categories: small, medium, and large. As can be seen in Table 6.7, certain size thresholds are chosen empirically to split the dataset, giving a nearly balanced number of cases in each size category. Then, we repeat the same experiments described in section 6.4.1.2 using the ULBP based on the ribbon sectors on the subset of the dataset of different size categories.

Table 6.7: Division of the dataset (395) into three lesion size categories.

| Size Category | Threshold (pixels) | No. of Cases | Regular | Irregular |
|---|---|---|---|---|
| Small | < 4900 | 128 | 53 | 75 |
| Medium | < 16000 | 137 | 49 | 88 |
| Large | >= 16000 | 130 | 79 | 51 |

As an exception, we are using protocol1, which uses five-fold cross-validations to evaluate this method since it is difficult to determine how many images in each size category were separated initially for internal testing in protocol3. Furthermore, protocol3 requires recalculating the ratios of the number of cases in each size category for training, evaluation, and testing sets each time a model is trained with different size categories.

Table 6.8 below shows the results of three ULBP models trained on different size category subsets of the internal dataset. Each model outperforms the model trained on the whole dataset (see section 6.4.1.2) by around 5%, 4%, and 3% for the small, medium, and large models. The gap between sensitivity and specificity in the case of the small model is minimal at around 4%, and it is very high in the other two models at about 30%. This behaviour could be traced back to the fact that most cases have small sizes (see Figure 6.12).

Table 6.8: ULBP method trained and tested on three subsets of the dataset (small, medium, and large-size nodules). (see Appendix B.5 Table 7)

| Small Size<br>ULBP (58), SVM-poly | | | |
|---|---|---|---|
| Margin | Acc | Reg | Irreg |
| 10 | 75 | 73 | 76 |
| 11 | 77 | 75 | 79 |

| Medium Size<br>ULBP (58), SVM-poly | | | |
|---|---|---|---|
| Margin | Acc | Reg | Irreg |
| 10 | 76 | 53 | 89 |
| 14 | 76 | 57 | 86 |

| Large Size<br>ULBP (58), SVM-poly | | | |
|---|---|---|---|
| Margin | Acc | Reg | Irreg |
| 13 | 75 | 89 | 53 |
| 17 | 75 | 90 | 51 |

We additionally split the internal dataset into small and large categories to further investigate the effect of the lesion sizes on the method's performance. Table 6.9 shows two category splits, small and large, using a size threshold of 6000 pixels.

Table 6.9: Division of the dataset (395) into two lesion size categories.

| Size Category | Threshold (pixels) | No. of Cases | Regular | Irregular |
|---|---|---|---|---|
| Small | < 6000 | 151 | 62 | 89 |
| Large | >= 6000 | 244 | 119 | 125 |

The external dataset is split using the same splitting threshold (6000 pixels) used in the internal dataset to test the model based on lesion size on the external testing set. Table 6.10 below shows the number of cases of the two classes for each doctor's class labels (A, B, and C).

Table 6.10: External dataset splits into small and large categories using three doctors' labels. Size Category and Threshold measured in pixels.

| Size Category | Threshold | No.Cases | Reg.A | Irreg.A | Reg.B | Irreg.B | Reg.C | Irreg.C |
|---|---|---|---|---|---|---|---|---|
| Small | < 6000 | 40 | 22 | 18 | 15 | 25 | 20 | 20 |
| Large | >= 6000 | 60 | 22 | 38 | 25 | 35 | 29 | 31 |

Tables 6.11 and 6.12 below show the ULBP method based on ribbon sectors trained and tested on small and large lesion categories. The method is tested on internal and external testing sets using protocol1. The method performs better than the three category models described before when tested on the internal testing set. The method gives the highest accuracies of 76% and 73 % and smaller gaps between sensitivity and specificities of 4 and 1% for small and large models, respectively. The small model's performance on the external testing set is quite low. In contrast, the large model's performance is significantly higher, achieving an accuracy of 77% using doctor C's ground truth with a gap between sensitivity and specificity of 13%.

Table 6.11: ULBP method trained and tested on a subset of the dataset (small-size nodules). (see Appendix B.5 Table 8)

| **Small Size**<br>**ULBP (58), protocol1**<br>SVM (C=100, degree=1, gamma=0.1, kernel='rbf') | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | |
| Margin | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 11 | 76 | 74 | 78 | 52 | 42 | 64 | 54 | 41 | 62 | 54 | 43 | 65 |
| 15 | 72 | 66 | 75 | 55 | 44 | 68 | 58 | 45 | 66 | 60 | 48 | 71 |

Table 6.12: ULBP method trained and tested on a subset of the dataset (large-size lesions). (See Appendix B.5 Table 9)

| **Large Size**<br>**ULBP (58), protocol1**<br>SVM (C=100, degree=1, gamma=0.1, kernel='rbf') | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | |
| Margin | Acc. | Reg. | Irreg. | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 16 | 70 | 70 | 71 | 74 | 73 | 75 | 72 | 67 | 75 | 77 | 70 | 83 |
| 21 | 73 | 73 | 74 | 71 | 67 | 73 | 70 | 65 | 74 | 74 | 66 | 81 |

#### 6.4.1.7 Training one Model using Different Size Dependent Ribbon Widths

From the experiments presented above, we can conclude that the size of the lesions affects the models' performance. As a second attempt to tackle this issue, we investigate using different ribbon widths for different categories of input image sizes to train one model. Table 6.11 above shows that a ribbon width of 11 pixels on the small-size nodules and 21 pixels on the large-size nodules gives the best accuracies on internal tests. Therefore, using these two ribbon widths to train one model is worth attempting. For

comparison, we use other ribbon widths in combination with different split thresholds for the size categories presented in Table 6.13 below, confirming that the two chosen ribbon widths are performing the best. The method achieves the highest accuracy and the smallest gap between sensitivity and specificity of 74% and 1%, respectively, which is slightly better than the original method's performance of 72% (see Table 6.2).

Table 6.13: Using different ribbon widths depending on the input nodule size to train ULBP sector-based model using internal dataset.

| ULBP (58), SVM-poly protocol1 | | | |
|---|---|---|---|
| **Margin** | **Acc** | **Reg** | **Irreg** |
| **small (11)/ large (21) thresh(6000)** | **74** | **74** | **73** |
| small (16)/ large (18) thresh(6000) | 70 | 67 | 72 |
| small (16)/ large (18) thresh(7000) | 70 | 68 | 71 |
| small (16)/ large (18) thresh(5000) | 70 | 67 | 71 |
| small (11)/ large (21) thresh(5000) | 73 | 72 | 74 |

All the results in this subsection demonstrate the viability of using ULBP texture feature vectors extracted from lesion border ribbons for lesion border irregularity recognitions, notwithstanding the absence of widely agreed class labelling systems. The sector-based ULBP schemes show improved effectiveness over the corresponding ULBP scheme extracted from the whole ribbon. We investigated the effect of the lesion sizes on the choice of ribbon width with improvements of different significance.

### 6.4.2 HOG based Method

Since the ribbon-based ULBP method using sectors achieved the best performances, we attempt only this approach in our HOG-based method. Also, the same evaluation protocol3 is used in the experiments for consistency. Like the ULBP method, we use nine sectors of equal radial distances, which might have different shapes and sizes depending on the lesion's shape. Further, the two approaches of radial distances and morphological operations are used to build the ribbon; however, both approaches give similar performances. Therefore, like in the case of the ULBP method, we present the results using the morphological ribbon construct for consistency.

Table 6.14 presents the performance of the trained HOG models on the internal, external, and external agreed testing sets. Methods performance is overall lower than those achieved by the ULBP-based method reaching the highest accuracy of 68% on the internal testing and 67% on external testing on doctor B's ground truth. The performances on the external agreed cases are higher but still much lower than that of the ULBP method.

Table 6.14: Experimental results of the HOG method based on ribbon using nine sectors on internal and external testing sets. (See Appendix B.6 Table 10)

| **HOG ribbon sectors** SVM(C=1, degree=5, gamma=1, kernel='poly') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Margin | Acc | Reg | Irreg | Acc | Acc | Reg | Irreg | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 21 | 68 | 57 | 77 | 57 | 36 | 73 | 59 | 38 | 73 | 58 | 39 | 76 | 62 | 37 | 79 |
| 24 | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 | 67 | 47 | 81 |

### 6.4.3 HOL based Method

The HOL method is already used in our first case study of abnormality recognition in building materials. We conducted experiments to test the performance of using HOL for lesion border irregularity recognition extracted from partitioned sectors of nodule ribbons and concatenated. First, the ED edge detector described in section 3.3.1 is used to extract the nodule edges from the ribbon area. Then, the histogram of linearity is calculated from the edge segments and used to classify regular and irregular nodule borders. Table 6.15 presents three experiments conducted using the whole ribbon and ribbon sectors using two different classifiers kNN and SVM. As in the previous experiments, a range of ribbon widths is used to determine the optimal one, and the same nine sectors are used.

Table 6.15: Whole ribbon (morphological) using protocol3. (See Appendix B.7 Table 11)

| **HOL Whole Ribbon** SVM(kernel='rbf') **Internal Testing** | | | |
|---|---|---|---|
| Margin | Acc. | Reg. | Irreg. |
| 13 | 53 | 54 | 52 |
| 21 | 57 | 59 | 55 |

| **HOL Ribbon Sectors** SVM( kernel='rbf') **Internal Testing** | | | |
|---|---|---|---|
| Margin | Acc. | Reg. | Irreg. |
| 19 | 55 | 56 | 55 |
| 21 | 55 | 51 | 59 |

| **HOL Ribbon Sectors** (kNN k=1) **Internal Testing** | | | |
|---|---|---|---|
| Margin | Acc. | Reg. | Irreg. |
| 10 | 60 | 58 | 62 |
| 21 | 53 | 53 | 52 |

The results show 60% accuracy at best when the kNN classifier is used. These disappointing results demonstrate that the HOL does not have any discriminating power in relation to thyroid nodule border irregularity. This may be related to two issues; (1) The ED algorithm is unable to extract most of the edges that contributed to the border irregularity due to poor grayscale intensities in US images (2) it could also be attributed to the fact that the edge detection algorithm extracts edges that are unrelated to the border but reflect the nature of the tissue within the ribbon. In contrast, the detected edges from glass facade panels often appear in the vicinity of the abnormal texture features, including crack components. Accordingly, we did not attempt to expand these experiments to include tests on external datasets.

### 6.4.4 CNN based Method

In the following, we present the experimental results of the CNN methods based on VGG16 and ResNet50 for border irregularity recognition. The models are trained using transfer learning by fine tunning described before. Since training CNN models take much longer than other traditional methods

presented previously, training 100 models, as is the case in using protocol3 for evaluation, is unrealistic. Therefore, we adapted five-fold cross-validation (protocol1 section 5.5.1.1), producing five models when trained on the internal dataset. Then the trained models are used to test the external dataset. Also, we use only two ribbon widths of 12 and 18 pixels due to longer training times. The configuration of all models was as follows:

MiniBatchSize = 8, MaxEpochs = 45, InitialLearnRate =1e-4

Tables 6.16 and 6.17 show the experimental results using the two CNN architectures using the two ribbon widths. The accuracy, specificity, and sensitivity of individual folds and the average of the five folds are recorded for internal, external, and external agreed cases. However, we present the average accuracies here and the full five-fold results in the attached Appendix B.

Table 6.16: VGG16 model using ribbon. (See Appendix B.8 Tables 12 and 13)

| **VGG16** Average of Five-fold | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Margin | Acc. | Reg. | Irreg. | Acc | Acc | Acc | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 12 | 89 | 91 | 87 | 80 | 81 | 80 | 75 | 74 | 76 | 82 | 85 | 78 | 87 | 91 | 81 |
| 18 | 89 | 89 | 89 | 82 | 81 | 84 | 75 | 73 | 79 | 83 | 84 | 81 | 88 | 91 | 85 |

Table 6.17: ResNet50 model using ribbon. (see Appendix B.8 Tables 14 and 15)

| **ResNet50** Average of Five-fold | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Margin | Acc. | Reg. | Irreg. | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 12 | 85 | 83 | 88 | 78 | 71 | 87 | 71 | 64 | 83 | 79 | 74 | 84 | 84 | 80 | 88 |
| 18 | 84 | 83 | 84 | 78 | 77 | 79 | 76 | 74 | 80 | 84 | 85 | 82 | 87 | 90 | 83 |

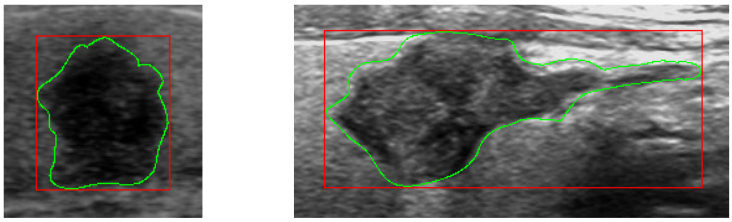
From the results above, we can observe similar patterns as in the handcrafted feature-based methods. In general, the accuracy of internal testing is better than external testing; however, external testing performance using doctor C's label is usually higher than the other two doctors' labels, and the best performance is achieved on external agreed cases. This might be explained by doctor C being more experienced than the other two. VGG16 architecture performs slightly better than ResNet50, reaching the highest accuracies of 89%, 83%, and 88% on internal, external, and external agreed cases. In both models, the ribbon width of 18 performs better than 12, although, in the case of ResNet50, the ribbon width of 18 gives 1% lower accuracy than 12 on internal testing; however, it has a much lower gap between sensitivity and specificity and higher performances on external testing. The better performance of wider ribbon widths can be explained by covering wider border regions. On the other hand, the slightly better performance of the VGG16 model might be due to the shallower architecture of VGG16 compared to ResNet50, which captures the US image's poor intensities better.

## 6.5 Result Analyses and Discussions

In the following, we discuss some attempts for irregularity recognition employing the entire bounding box for ULBP feature extraction or some input image pre-processing to improve performance, along with a comparison of the proposed methods.

### 6.5.1 Irregularity Recognition based on the Lesion Bounding Box

In our first attempt to apply ULBP to the irregularity classification, the simple bounding box around the ROI points is used for texture analyses. All the pixels inside the bounding box (red box Figure 6.13) are used for extracting ULBP codes. A histogram of the ULBP is formed, normalized, and fed to a classifier for regular and irregular border classification. Depending on the lesion's shape, the bounding box could involve fewer (Figure a) or more regions (Figure b) from outside the border of the lesion. Although this method could give reasonable accuracies, it could not reflect the actual border irregularity due to the involvement of extra textures from inside and outside the lesion, which may have other indications than the border irregularity. The approach performed poorly in the few experiments on the internal dataset, as shown in Table 6.18 below.



(a) (b)

Figure 6.13: ULBP based on the bounding box for lesion border irregularity recognition.

Table 6.18: Results of ULBP method based on bounding box on internal testing using protocol3.

| ULBP (58) Bounding Box Internal Dataset | | | |
|---|---|---|---|
| **Classifier** | **Acc** | **Reg** | **Irreg** |
| kNN (k=13) | 56 | 61 | 52 |
| SVM (poly-deg=7) | 61 | 63 | 59 |

### 6.5.2 Image Pre-processing for Texture-based Methods

All the methods proposed in this thesis rely on the raw input image, i.e., no pre-processing of the images in terms of sharpening or noise reduction has been used. Any potential pre-processing operations will not affect the methods based on the ROI points described in chapter 5 since they do not consider the pixel intensities in their assessments. However, the pre-processing may improve the performance of the other methods, such as ULBP, HOG, and CNN, which assess the pixel intensities around the borderline. Some initial sharpening and blurring filters based on kernels of sizes three, five, and seven pixels have

been attempted; however, they led to lower performances. The use of speckle noise reduction filters, which have been reported in the literature, can provide another intriguing pre-processing strategy [177].

### 6.5.3 Comparing and Analysing all Methods

In the following, we present the comparison and analyses of the experimental results of all methods proposed in this chapter based on texture analysis (pixel intensities) around the borderline. For easier comparison, the best-performing configuration is selected from each method. The highest overall accuracy and the smallest gap between sensitivity and specificity are considered for the selection. Both methods' performance on the internal and external testing sets using individual and agreed class labels are presented in the comparison. Protocol3 is used in most of the methods with fixed randomization, ensuring that the same splits of the training/evaluation sets are used when 100 models are trained. Some other methods use five-fold cross-validations, including CNN-based methods. As in chapter 5, we use the colours given in Table 5.30 to show different rankings of the best-performing methods, where red, green, and blue represent the first, second, and third best-performing methods. Finally, we introduced different methods abbreviations shown in Table 6.19 in the comparison for easier comparison and to save space.

Table 6.19: Methods abbreviations.

| Abbreviation | Description |
|---|---|
| **WolRbn** | Whole Ribbon |
| **RbnSec** | Ribbon Sectors |
| **Rad** | Ribbon based on radial distances |
| **Morp** | Ribbon based on Morphological dilation and erosion |
| **Gmw** | Generic Margin Width |
| **CnnVgg16** | CNN based on VGG16 |
| **CnnResNet50** | CNN based on ResNet50 |

Table 6.20: Comparison of the best-performing methods based on the texture analysis, including CNN models.

| Method | Classifier (Ribbon width) | Internal Dataset | | | External Dataset | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| ULBP (**WolRbn**.**Rad**) | SVM (18) | 65 | 65 | 66 | 56 | 43 | 66 | 56 | 43 | 65 | 51 | 39 | 63 |
| ULBP (**WolRbn**.**Morp**) | SVM (10) | 65 | 63 | 67 | 57 | 52 | 61 | 61 | 57 | 63 | 54 | 49 | 59 |
| ULBP (**RbnSec.Morp**) | SVM (22) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 |
| ULBP(G3) (**RbnSec.Morp**) | SVM (11) | 72 | 72 | 72 | 64 | 52 | 73 | 60 | 48 | 68 | 61 | 49 | 73 |
| ULBP (**RbnSec**.**Gmw**) | **Model1** | 65 | 69 | 62 | 74 | 70 | 77 | 74 | 72 | 75 | 77 | 71 | 82 |
| ULBP (**RbnSec**.**Gmw**) | **Model2** | 68 | 70 | 66 | 69 | 64 | 73 | 69 | 65 | 72 | 72 | 65 | 78 |
| HOG (**RbnSec**.**Morp**) | SVM (24) | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 |
| Cnn**Vgg16** | Ribbon(18) | 89 | 89 | 89 | 82 | 81 | 84 | 75 | 73 | 79 | 83 | 84 | 81 |
| Cnn**ResNet50** | Ribbon(12) | 84 | 83 | 84 | 78 | 77 | 79 | 76 | 74 | 80 | 84 | 85 | 82 |

Table 6.20 shows that the two CNN methods based on VGG16 and ResNet50 perform similarly and outperform all other handcrafted feature-based methods on both internal and external testing sets. Among handcrafted feature-based methods, the method based on ULBP(G3) (**RbnSec.Morp**) using ribbon sectors with a ribbon width of 11 pixels performs the best on the internal dataset achieving

accuracy, sensitivity, and specificity of 72%, 72%, and 72%, respectively performing slightly better than ULBP(58) using all uniform codes, but it has lower performances on external testing. The ULBP Generic Margin Width (**RbnSec.Gmw**) based method using **Model2** performs the next best on internal testing. On the external testing, ULBP (**RbnSec.Gmw**) method using **Model1** performs the best, achieving 74%, 74%, and 77% for doctors A, B, and C, respectively, with a small gap between the sensitivity and specificity. The HOG (**RbnSec.Morp**) based method ley behind ULBP, achieving 66% overall accuracy, with the ULBP method based on the whole ribbon (no sectors) (**WolRbn.Rad**) performing the worse on the external dataset.

Table 6.21: Methods performance comparison when tested on the external set using agreed ground truth.

| Method | Classifier | Internal Testing | | | External Testing Agreed Cases | | |
|---|---|---|---|---|---|---|---|
| | | Acc | Reg | Irreg | Acc | Reg | Irreg |
| ULBP (**RbnSec.Morp**) | SVM (17) | 64 | 65 | 64 | 77 | 70 | 81 |
| ULBP (**RbnSec.Gmw**) | **Model1** | 65 | 69 | 62 | 82 | 73 | 88 |
| ULBP (**RbnSec.Gmw**) | **Model2** | 68 | 70 | 66 | 75 | 70 | 79 |
| HOG (**RbnSec.Morp**) | SVM (10) | 59 | 52 | 66 | 73 | 53 | 86 |
| Cnn**Vgg16** | Ribbon(12) | 85 | 91 | 77 | 88 | 91 | 85 |
| Cnn**ResNet50** | Ribbon(18) | 79 | 80 | 77 | 87 | 90 | 83 |

Table 6.21 above compares the method's performances on the external dataset with agreed class labels. The CNN-based methods again perform the best, achieving the highest overall accuracy, specificity, and sensitivity of 89%, 93%, and 83% using VGG16, with ResNet50 lying slightly behind. Among the handcrafted feature-based methods, again ULBP (**RbnSec.Gmw**) method using **Model1** performs the best, achieving overall accuracy, specificity, and sensitivity of 82%, 73%, and 88%, respectively. The ULBP (**RbnSec.Morp)** method performed the second best, achieving 77% overall accuracy.

## 6.6 Summary

In this chapter, we investigated various aspects of designing and evaluating the performance of automatic border irregularity recognition of thyroid cancer nodules using several versions of known texture features extracted from the ribbon region surrounding the ROI points along the interpolated lesion border. Two different methods based on radial distances and morphological erosion and dilations are used to construct ribbons of different widths. We have shown the correlations between the lesion sizes and the chosen ribbon width by splitting the data into several size categories and training ULBP models separately. Also, training the ULBP model using several ribbon widths of the same image showed benefits, at least in the case of external testing. Although the black box nature of the CNN-based models is not quite acceptable to clinicians, it performs superior to the handcrafted methods. However, the CNN models were not as good as the other handcrafted-based methods (see chapter 5) based on morphological features extracted from the approximated border from the ROI point. In

addition, we have shown that the HOG method is not giving reasonable performances, and the HOL using an edge detector is not promising for texture-based border irregularity recognition.

The experimental results were promising but not as good as those achieved based on the border interpolated curve in chapter 5. It confirmed the viability of using the various texture analysis of lesion border ribbons for recognizing border irregularity. A question arises: What do all these methods collectively inform us on how to deal with the problem of lesion border abnormality? Next, in chapter 7, we attempt to shed some light on this question by conducting several experiments on fusion schemes of various collections of the developed schemes for border irregularity recognition.

# Chapter 7: Combining Multi-Classifiers for Lesion Border Irregularity

In chapters 5 and 6, various methods were introduced using two different approaches for classifying and assessing the border irregularity of the thyroid cancer nodules. The first approach uses a finite set of ROI points marked by the doctors on the nodule border to interpolate a nodule border closed curve to develop irregularity recognition schemes by estimating its FD or analysing distance functions to known regular reference shapes related to the ROI points. The second approach considers the pixel intensities and textures around the interpolated borderline. Our various proposed methods (1) use two main approaches based on borderline, and texture analysis and (2) borderline analysis-based features extract morphological features of different natures using border distances or FD in addition to analysis of border distances in spatial and frequency domains and by using TDA and (3) the methods had different levels of success, ranging from very modest (around 65%) to considerable (acceding 80%). All the above differences make our feature schemes diverse enough to consider investigating different ways of combining several of these schemes and determine if it is possible to improve the accuracy of classification prediction over and above the best-performing single scheme(s). There are different known strategies for combining multi-classifier, but due to the fact that we have developed several schemes that differ in many ways, it is sensible to limit the scope of investigations in this chapter to select and use a few applicable fusion methods, such as decision and score level fusion as well as combining the decisions using DT. We did not attempt to use the CNN-model decisions and scores in the fusion schemes due to their already high performances and the difficulty in determining the same testing sets as other methods because they use different evaluation protocols.

For our fusion schemes, we select a few best-performing methods from each group of methods. As criteria for the selection, the highest accuracy and smallest gap between sensitivity and specificity on internal and external testing sets are used, although more focus was on internal testing. Then, the recorded class predictions and probabilities or scores associated with the two classes are used for different fusion schemes or combined in a DT.

## 7.1 Introduction

Fusing (ensembling) multiple classifiers is a well-understood method of combining multiple classification schemes to analyse a given cloud point of data records. Many methods of ensembling multiple classifiers exist [178]; however, we adopt simple score and decision-level fusion. The classifiers must be accurate and diverse for the fusion to perform better than the individual ones. Our methods are diverse and have relatively high accuracies beyond random guesses. In the following, we describe some of the various existing applications of the fusion schemes.

Different morphological features, such as roughness and ellipticity, are combined at the feature level with lesion edge features, such as curvature, to improve breast cancer malignancy recognition [179]. Simple decision-level fusion by majority voting of several methods, such as ULBP, Gabor filters, and statistical moments are used for ovarian cancer recognition [104]. A score and decision level fusion of three methods based on LBP, Histogram of grayscale, and statistical moments are used for nasopharyngeal carcinoma cancer recognition, where the score level fusion outperformed the decision level fusion [180].

Fusion can be applied at different levels, including:

1. **The feature level** - Here, feature vectors from the constituent classifiers are combined by concatenation with or without normalization, and the resulting feature vectors will be trained and tested using a chosen classifier. This approach is suitable when the feature vectors are less correlated to each other with respect to different classes, but this is difficult to achieve when the number of training samples is relatively small.

2. **The decision level** – the decisions of the constituent classifiers are combined to make new decisions using specific criteria. Different criteria are used in the literature, including the weighted majority rule, where the weights are determined by the performance of the fused schemes at the training stage. We shall investigate the ***simple majority rule*** whereby all classifiers are given the same weights, and this approach can only be used with an odd number of classification schemes.

3. **The score level** – Here, each of the constituent classifiers outputs a score with each class, and the final outcome from testing an input image will be the class with the maximum added scores. The individual scores associated with each classifier are often expressed as probability values that indicate a kind of confidence in the decision made by the corresponding classifier.

## 7.2 Proposed Fusion schemes and Experimental Results

In the following, we present the results of decision level (majority rule) and score level fusion of various combinations of two, three, and five methods from both ROI and ribbon-based texture analysis schemes. The decision fusion using a majority rule requires an odd number of methods; hence the use of three and five methods and for consistency used in other schemes of score level fusion and methods combination in DT. We also use the minimum required method numbers of two for the later two schemes. The feature level fusion is not attempted due to the different lengths and scales of the feature vectors used in the various methods. In addition, the HOL texture analysis scheme applied to the lesion border ribbon did not achieve any discriminating power; therefore, we decided not to include it in the fusion or the DT scheme.

For consistency, we use the same internal testing splits made by protocol3 (see section 5.5.1.1) previously used to evaluate all our proposed methods, i.e., the same 118 internal testing cases from the internal dataset of 395 cases are used. The same fusion schemes are applied to the external testing set of 100 images using individual and agreed doctor's class labels. Note that we already trained and tested each participating scheme on the internal datasets and tested on the external dataset in chapters 5 and 6, and experiments associated a decision and a score with each image. The selected fusion schemes are evaluated first on the internal 118 testing set and then applied to the 100 images of the external testing set. In each experiment, the accuracy (**Acc**), specificity (**Reg**) and sensitivity (**Irreg**) are recorded.

For simplicity and to save space, we use abbreviations of the methods name shown in Table 7.1 below to present the results of the fusion schemes.

Table 7.1: Method name abbreviations for fusion schemes.

| **Abbreviation** | **Description** |
|---|---|
| Dist | Distances |
| Cent | Centroid |
| Ellip | Fitted Ellipse |
| Gaus | Fitted Gaussian |
| Conv | Convex Hull |
| ConvEllip | Convex Fitted Ellipse |
| Ftt | FFT |
| TDA | TDA |
| FD | Fractal Dimensions |
| RibSec | Ribbon Sectors based |
| Ang | Equi-angdisp sampling method |
| Arc | Equi-arclength sampling method |
| Morph | Ribbon based on morphological operations |
| Rad | Ribbon based on radial distances |

### 7.2.1 Decision Level Fusion (Majority Rule)

In the case of decision fusion of the proposed methods, their classification predictions for each testing case are fused using the majority rule. The majority rule always requests the predictions (decisions) to be an odd number. In the following, we present several scenarios of fusing some of the best-performing methods from both ROI and texture analysis approaches. The fusion is conducted on the predictions of internal (118 images) and external (100 images) testing sets using three and five best-performing methods.

#### 7.2.1.1 Three-Methods Fusion

Tables 7.2 to 7.5 present fusion results using three methods. On the internal testing set, the fusion does not improve the accuracy; it even deteriorates slightly in the case where the HOG method is used in the fusion (see Table 7.5). On the other hand, there are significant improvements in the accuracy of the

external testing set using individual and agreed class labels. However, it worsens when the HOG method is again used in the fusion.

Table 7.2: Decision-based fusion of three methods. (ConvEllip(Ang), FD, FFT).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Ang. | Svm (rbf) | 83 | 80 | 86 | 76 | 77 | 75 | 74 | 78 | 72 | 83 | 82 | 84 | 85 | 80 | 88 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Decision Fusion | | 83 | 85 | 81 | 82 | 84 | 80 | 78 | 83 | 75 | 91 | 90 | 92 | 93 | 87 | 98 |

Table 7.3: Decision-based fusion of three methods. (ConvEllip(Arc), FD, FFT).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Decision Fusion | | 83 | 81 | 83 | 80 | 83 | 86 | 80 | 77 | 83 | 73 | 92 | 92 | 93 | 90 | 95 |

Table 7.4: Decision-based fusion of three methods. (ConvEllip(Arc), FD, ULBP).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Decision Fusion | | 83 | 81 | 84 | 82 | 82 | 82 | 78 | 80 | 77 | 89 | 86 | 92 | 92 | 87 | 95 |

Table 7.5: Decision-based fusion of three methods. (ConvEllip(Ang), HOG, ULBP).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| HOG (RibSec.) Morph. | Svm (poly) | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 | 67 | 47 | 81 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Decision Fusion | | 74 | 74 | 73 | 66 | 34 | 91 | 72 | 40 | 93 | 69 | 39 | 98 | 85 | 67 | 98 |

#### 7.2.1.2 Five-Methods Fusion

Tables 7.6 to 7.9 presents the decision fusion results using predictions of five methods. This fusion improves the accuracy of internal testing by about 2%. The improvement in accuracy is 2% and 3% on external testing using individual and agreed doctor's labels. Interestingly, the internal accuracy is not deteriorating when HOG is used in the fusion. It even improves the external accuracy by 2% (see Table 7.9). This is due to involving more high-performing methods in the fusion.

Table 7.6: Decision-based fusion of five methods. (ConvEllip(Ang), FD, FFT,Conv, TDA).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Ang. | Svm (rbf) | 83 | 80 | 86 | 76 | 77 | 75 | 74 | 78 | 72 | 83 | 82 | 84 | 85 | 80 | 88 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| TDA, Dist (Conv.) Ang. | Svm (rbf) | 81 | 81 | 80 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| Decision Fusion | | 86 | 93 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 88 | 92 | 84 | 92 | 93 | 91 |

Table 7.7: Decision-based fusion of five methods. (ConvEllip(Arc), FD, FFT,Conv, TDA).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| TDA, Dist (Conv.) Ang. | Svm (rbf) | 81 | 81 | 80 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| Decision Fusion | | 85 | 91 | 80 | 83 | 91 | 77 | 77 | 88 | 70 | 90 | 94 | 86 | 93 | 97 | 91 |

Table 7.8: Decision-based fusion of five methods. (ConvEllip(Arc), FD, ULBP,Conv, TDA).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| TDA, Dist (Conv.) Ang. | Svm (rbf) | 81 | 81 | 80 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| Decision Fusion | | 86 | 93 | 81 | 85 | 91 | 80 | 77 | 85 | 72 | 90 | 92 | 88 | 93 | 93 | 93 |

Table 7.9: Decision-based fusion of five methods. (ConvEllip(Arc), FD, ULBP,Conv, HOG).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| HOG (RibSec.) Morph. | Svm (poly) | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 | 67 | 47 | 81 |
| Decision Fusion | | 84 | 83 | 84 | 83 | 86 | 80 | 79 | 85 | 75 | 90 | 90 | 90 | 93 | 90 | 95 |

Generally, our observations from decision-based fusion experiments are that the fusion of three methods does not improve the overall performance and, in some cases, deteriorates. In contrast, the performance increases by up to 5% on the external testing set. Furthermore, using five classifiers in the fusion performs significantly better than using three classifiers. The HOG method does not contribute to the accuracy improvement in the fusion.

### 7.2.2 Score Level Fusion (Score Averaging)

Since score level fusion can be done on any number of classifiers, we choose three and five classifiers for consistency, as the same number of classifiers were used in decision fusion methods. In addition, we use the minimum required of two classifiers for score-level fusion to see its effect on performance improvement. The method simply adds the scores corresponding to each of the two classes for all involved methods and assigns the label of the class having the highest total score to the input image.

#### 7.2.2.1 Two-Methods Fusion

Tables 7.10 and 7.11 shows the results of fusing the two methods' scores. In the first table, we fuse two methods from the texture analysis approach for thyroid nodule border irregularity using HOG and ULBP methods. The method performance is deteriorating on all testing sets. On the other hand, when the two best methods from the borderline analysis approach are fused, we observe accuracy improvements of up to 3% on internal testing and marginal improvement on external testing, even though only two methods are used (see Table 7.11).

Table 7.10: Score-based fusion of two methods. (HOG and ULBP).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| HOG (RibSec.) Morph. | Svm (poly) | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 | 67 | 47 | 81 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Score Fusion | | 65 | 55 | 73 | 63 | 52 | 70 | 62 | 51 | 73 | 65 | 55 | 73 | 70 | 60 | 77 |

Table 7.11: Score-based fusion of two methods. (ConvEllip and FD).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Ang. | Svm (rbf) | 83 | 80 | 86 | 76 | 77 | 75 | 74 | 78 | 72 | 83 | 82 | 84 | 85 | 80 | 88 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| Score Fusion | | 86 | 81 | 91 | 81 | 80 | 82 | 79 | 80 | 78 | 86 | 82 | 90 | 90 | 83 | 95 |

#### 7.2.2.2 Three-Methods Fusion

Four combinations of three methods from both border irregularity analysis approaches are selected for score-level fusion in Tables 7.12 to 7.15. The results show the same accuracy improvements on the internal testing as in the two-methods fusion; however, the gap between the specificity and sensitivity is smaller (see Table 7.12). The method also shows up to 4% improvements in external testing (doctor B Table 7.14). Methods performance using ULBP is slightly better; in contrast, the HOG method does not improve the performance (see Tables 7.14 and 7.15)

Table 7.12: Score-based fusion of three methods. (ConvEllip(Ang), FD, FFT).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Ang. | Svm (rbf) | 83 | 80 | 86 | 76 | 77 | 75 | 74 | 78 | 72 | 83 | 82 | 84 | 85 | 80 | 88 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Score Fusion | | 86 | 85 | 88 | 82 | 80 | 84 | 78 | 78 | 78 | 87 | 82 | 92 | 90 | 80 | 98 |

Table 7.13: Score-based fusion of three methods. (ConvEllip(Arc), FD, FFT).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Score Fusion | | 85 | 83 | 86 | 82 | 80 | 84 | 78 | 78 | 78 | 87 | 82 | 92 | 90 | 83 | 95 |

Table 7.14: Score-based fusion of three methods. (ConvEllip(Arc), FD, ULBP).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Score Fusion | | 84 | 80 | 88 | 83 | 82 | 84 | 81 | 82 | 80 | 88 | 84 | 92 | 92 | 83 | 98 |

Table 7.15: Score-based fusion of three methods. (ConvEllip(Arc), HOG, ULBP).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| HOG (RibSec.) Morph. | Svm (poly) | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 | 67 | 47 | 81 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Score Fusion | | 79 | 76 | 81 | 79 | 73 | 84 | 77 | 72 | 80 | 84 | 76 | 92 | 88 | 77 | 95 |

#### 7.2.2.3 Five-Methods Fusion

The results of the five-methods score fusion are presented in Tables 7.16 to 7.19. Significant accuracy improvements of up to 6% can be observed from the results on all testing sets, reaching maximum accuracy of 96% on the external agreed cases. There are also significant improvements in the fusion involving ULBP and HOG; however, the improvements are lower when HOG is used in the fusion.

Table 7.16: Score-based fusion of five methods. (ConvEllip(Ang), FD, FFT, Conv, TDA).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Ang. | Svm (rbf) | 83 | 80 | 86 | 76 | 77 | 75 | 74 | 78 | 72 | 83 | 82 | 84 | 85 | 80 | 88 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| TDA, Dist (Conv.) Ang. | Svm (rbf) | 81 | 81 | 80 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| Score Fusion | | 90 | 91 | 89 | 87 | 89 | 86 | 81 | 85 | 78 | 92 | 90 | 94 | 96 | 93 | 98 |

Table 7.17: Score-based fusion of five methods. (ConvEllip(Arc), FD, FFT, Conv, TDA).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| TDA, Dist (Conv.) Ang. | Svm (rbf) | 81 | 81 | 80 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| Score Fusion | | 90 | 91 | 89 | 86 | 89 | 84 | 80 | 85 | 77 | 91 | 90 | 92 | 95 | 93 | 95 |

Table 7.18: Score-based fusion of five methods. (ConvEllip(Arc), FD, ULBP, Conv, TDA).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| TDA, Dist (Conv.) Ang. | Svm (rbf) | 81 | 81 | 80 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| Score Fusion | | 87 | 89 | 86 | 86 | 84 | 88 | 82 | 82 | 82 | 91 | 86 | 96 | 96 | 90 | 100 |

Table 7.19: Score-based fusion of five methods. (ConvEllip(Arc), FD, ULBP, Conv, HOG).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| HOG (RibSec.) Morph. | Svm (poly) | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 | 67 | 47 | 81 |
| Score Fusion | | 86 | 85 | 88 | 82 | 84 | 80 | 80 | 85 | 77 | 89 | 88 | 90 | 93 | 90 | 95 |

From the results above, we can generally observe that the more classifiers involved in the fusion, the higher the performance improvements. The fusion of 5 classifiers gives the best performance of 90%, 92%, and 96% for internal, external doctor C, and external agreed cases, respectively. All score-level fusion schemes show improvements in the performance of different significance even when only two methods are fused (see Table 7.11).

### 7.2.3 Decision Tree-Based Mining of Multi-Classifier

A Decision Tree (DT) is a standard datamining tool. In this section, we shall mine the various developed irregularity recognition schemes using the DT tool. It exploits the information gained from the decisions made by the constituent classifiers to construct a hierarchical classification scheme to prioritize testing the various schemes accordingly. It has obvious similarities with decision-based fusion, but it is more efficient at the testing stage, whereby it usually extracts fewer feature vectors in making its final decision. Our is not only to get better performances but to determine the priority of testing the different classifiers by ranking their performances.

Many methods exist for building a DT; however, they all share similar steps described in the following [181].

1- An attribute (feature) is selected from the training set to be the tree's root.
2- The training data is split into two subsets according to the selected attribute.
3- A new tree for each subset is constructed.
4- A link is created to connect the root of the current tree to each of the sub-tree roots and labelled with the attribute's value used to split the two subsets.
5- The same steps beginning from one are repeated to build each new subset tree until all branches reach a leaf, i.e., no more branching is possible.

The attribute selection step is the most crucial part of building the DT since it involves selecting the best attribute for splitting the data so that it assigns the maximum number of data samples to the correct classes. The most common measure for attribute selection is information gain, where an attribute is selected for splitting the data giving the highest information gain. Information gain can be measured either using entropy or the Gini index. Entropy is the amount of uncertainty in a set of data or events, where the higher the uncertainty of an event to happen, the higher its entropy, and consequently, the higher its information gain. The Gini impurity or Gini index is another measure for information gain and determines how good a split is in the DT. Entropy and Gini-index are both calculated from the probabilities of the individual classes in the dataset used for building the tree.

The training set is used to construct a DT, which is then used to predict the unseen testing data. DT is used as a classifier based on any discrimination features. However, we use the DT to combine the predicted decisions instead. DT can be built with any number of classifiers contrary to decision-based fusion. At each branch, the entire data is split in such a way that the information gain is maximised. The root node is associated with testing the best-performing classifier, and subsequently, at each subsequent node (branch), if the decisions agree with the agreed class label, then it is made into a leaf; otherwise, the arriving samples are split according to the decision of another classifier chosen again using

information gain. The process continues until all samples are classified. In binary classifiers, the constructed DT is a binary tree.

Although any odd or even number of classifiers from two can be used to build the DT, we attempt to use the same number of classifiers of two, three, and four for consistency with other fusion schemes. We again pick the same best methods and combine their decisions based on the DT classification. The constructed three using the internal testing decisions is then used to test the external testing set, i.e., we do not fit a new tree on the external testing; the fitted tree on the internal testing is used instead.

#### 7.2.3.1 Two-Methods DT

The following presents the results of combining two methods' decisions using DT and visualises some of the constructed DTs. The two methods based on ULBP and HOG from the texture analysis approaches do not show any performance improvements (see Table 7.20), similar to the previous fusion schemes. Using the two best methods from borderline analysis approaches shows improvements in the internal testing on the costs of a wider gap between specificity and sensitivity (see Table 7.21.

Table 7.20: Two methods DT. (HOG, ULBP)

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| HOG (RibSec.) Morph. | Svm (poly) | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 | 67 | 47 | 81 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Decision Tree | | 69 | 72 | 67 | 64 | 61 | 66 | 64 | 62 | 65 | 63 | 59 | 67 | 70 | 67 | 72 |

Table 7.21: Two methods DT. (ConvEllip(Ang), FD).

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Ang. | Svm (rbf) | 83 | 80 | 86 | 76 | 77 | 75 | 74 | 78 | 72 | 83 | 82 | 84 | 85 | 80 | 88 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| Decision Tree | | 86 | 76 | 94 | 80 | 75 | 84 | 78 | 75 | 80 | 85 | 78 | 92 | 89 | 77 | 98 |

The DT in Figure 7.1 below is constructed from the decisions in Table 7.21. The feature X1, which is the prediction made by the FD-inspired method, is chosen as the root attribute giving the highest information gain (Gini=0.496). As seen from the figure, for testing the external dataset, both features X0 and X1 are always needed, i.e., both classifier predictions are used to get to the final decision. In this case, the DT is not more efficient than a decision or score-based fusion regarding the number of classifiers needed to get to the final decision.

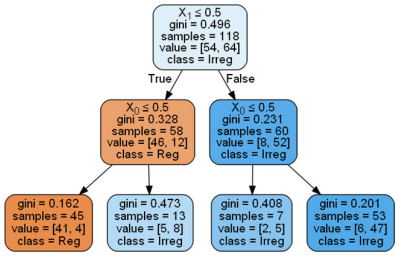



Figure 7.1: DT based on Table 7.21, feature(decisions) X0 = ConvEllip(Ang) and X1=FD.

### 7.2.3.2 Three-Methods DT

The results of the DT-based method using the three best classifiers are presented in Tables 7.22 to 7.25, and the best-performing DT method is visualized in the following. Combining decisions from three classifiers improves internal accuracy by 3%; however, it gives a higher gap between specificity and sensitivity (see Table 7.22).

Table 7.22: Three methods DT. (ConvEllip(Ang), FD, FFT)

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Ang. | Svm (rbf) | 83 | 80 | 86 | 76 | 77 | 75 | 74 | 78 | 72 | 83 | 82 | 84 | 85 | 80 | 88 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Decision Tree | | 86 | 76 | 94 | 80 | 73 | 86 | 76 | 70 | 80 | 87 | 78 | 96 | 89 | 77 | 98 |

It is interesting to see that the constructed DT picks FD-inspired (X1) again as the root attribute for the tree, although ConvEllip(Ang) (X0) has the same average accuracy but a higher gap between sensitivity and specificity. All three methods' predictions are needed to get to the final decision when the DT is used for testing, which means it is not more efficient than the other two fusion schemes regarding testing times or the number of involved classifiers.

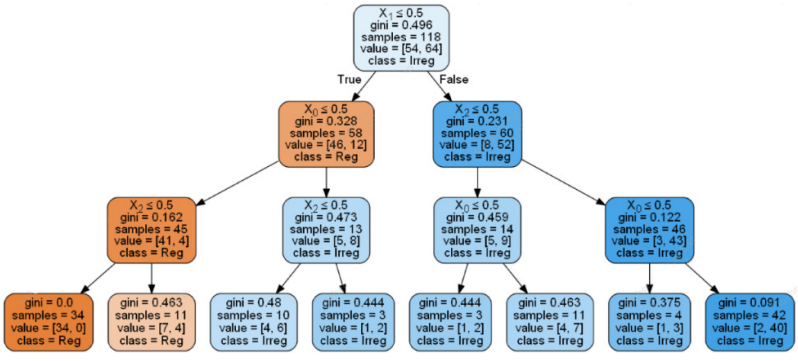



Figure 7.2: DT based on Table 7.22, feature(decisions) X0 = ConvEllip(Ang), X1=FD, and X2=FFT.

Table 7.23: Three methods DT. (ConvEllip(Arc), FD, FFT)

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Decision Tree | | 83 | 85 | 81 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 90 | 90 | 91 |

It is interesting to see that combining the three methods, including ULBP, slightly improves the accuracy of internal and external testing sets by around 1% and 2%, respectively (see Table 7.24).

Table 7.24: Three methods DT. (ConvEllip(Arc), FD, ULBP)

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Decision Tree | | 84 | 81 | 86 | 83 | 82 | 84 | 79 | 80 | 78 | 89 | 90 | 88 | 92 | 87 | 95 |

Table 7.25: Three methods DT. (ConvEllip(Arc), HOG, ULBP)

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| HOG (RibSec.) Morph. | Svm (poly) | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 | 67 | 47 | 81 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Decision Tree | | 79 | 85 | 73 | 82 | 89 | 77 | 77 | 88 | 70 | 89 | 92 | 86 | 92 | 93 | 91 |

Unlike the decision-level and score-level fusion, the DT-based scheme shows slight performance improvements even though the least performing methods, HOG and ULBP, are involved (see Table 7.25).

#### 7.2.3.3 Five-Methods DT

The following presents the final experiment results, which combine five classifiers using DT. All experiments improve accuracy on both internal and external testing sets of different significance. However, the combination of five methods, including ULBP, performs the best, achieving 90%, 90%, and 92% on internal, external doctor C, and external agreed class labels, respectively. Performance is greatly enhanced when FFT is replaced with the ULBP method, demonstrating that the ULBP method is better complemented with the methods based on interpolated borderline analysis (see Tables 7.27 and 7.28).

Table 7.26: Five methods DT. (ConvEllip(Ang), FD, FFT, Conv, TDA)

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Ang. | Svm (rbf) | 83 | 80 | 86 | 76 | 77 | 75 | 74 | 78 | 72 | 83 | 82 | 84 | 85 | 80 | 88 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| TDA, Dist (Conv.) Ang. | Svm (rbf) | 81 | 81 | 80 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| Decision Tree | | 88 | 96 | 81 | 79 | 89 | 71 | 75 | 88 | 67 | 84 | 90 | 78 | 88 | 93 | 84 |

Table 7.27: Five methods DT. (ConvEllip(Arc), FD, FFT, Conv, TDA)

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| FFT, Dist (Conv.) Ang. | Svm (rbf) | 78 | 80 | 77 | 75 | 77 | 73 | 73 | 78 | 70 | 86 | 86 | 86 | 85 | 80 | 88 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| TDA, Dist (Conv.) Ang. | Svm (rbf) | 81 | 81 | 80 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| Decision Tree | | 90 | 98 | 83 | 79 | 91 | 70 | 77 | 92 | 67 | 84 | 92 | 76 | 89 | 97 | 84 |

Table 7.28: Five methods DT. (ConvEllip(Arc), FD, ULBP, Conv, TDA)

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| TDA, Dist (Conv.) Ang. | Svm (rbf) | 81 | 81 | 80 | 82 | 82 | 82 | 74 | 75 | 73 | 81 | 78 | 84 | 88 | 83 | 91 |
| Decision Tree | | 90 | 98 | 83 | 81 | 91 | 73 | 77 | 90 | 68 | 90 | 96 | 84 | 92 | 97 | 88 |

As shown in figure 7.3 below, the more classifiers (methods) are used in constructing the DT, the more complex the tree's structure. In contrast to the previous DT based on two and three methods, we can see that it does not always need all five classifiers' predictions to predict the external testing labels. For the leaf shown by the red arrow (1), only three methods (X3, X1, and X4) are used, while the other two leaves (2) and (3) need four methods predictions. Further, the DT picks the method Conv (X3) as the root since it has the highest accuracy among the methods.

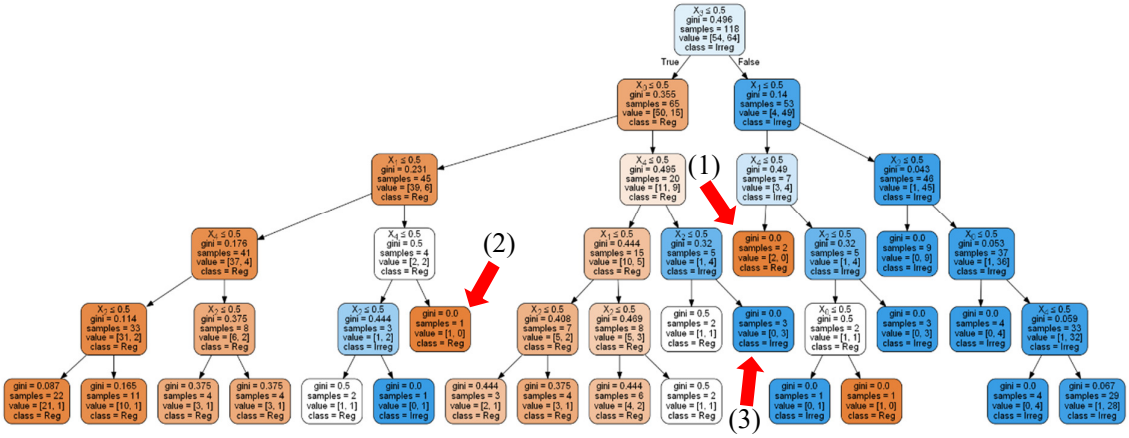



Figure 7.3: DT based on Table 7.28, features (decisions) X0 = ConvEllip(Arc), X1=FD, X2=ULBP, X3=Conv, and X3=TDA.

Table 7.29: Five methods DT. (ConvEllip(Arc), FD, ULBP, Conv, HOG)

| Method | Classifier | Internal Dataset | | | External Dataset | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| | | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Dist (ConvEllip.) Arc. | Svm (rbf) | 79 | 76 | 81 | 81 | 84 | 79 | 77 | 82 | 73 | 88 | 88 | 88 | 90 | 90 | 91 |
| FD Ang | Svm (rbf) | 83 | 85 | 81 | 81 | 89 | 75 | 77 | 88 | 70 | 86 | 90 | 82 | 90 | 90 | 91 |
| ULBP (RibSec.) Morph. | Svm (rbf) | 72 | 70 | 73 | 67 | 55 | 77 | 69 | 57 | 77 | 68 | 55 | 80 | 75 | 67 | 81 |
| Dist (Conv.) Ang. | Svm (rbf) | 84 | 93 | 77 | 75 | 89 | 64 | 73 | 90 | 62 | 84 | 94 | 75 | 85 | 93 | 79 |
| HOG (RibSec.) Morph. | Svm (poly) | 66 | 61 | 70 | 57 | 39 | 71 | 67 | 50 | 78 | 64 | 47 | 80 | 67 | 47 | 81 |
| Decision Tree | | 88 | 96 | 81 | 80 | 89 | 73 | 76 | 88 | 68 | 89 | 94 | 84 | 90 | 93 | 88 |

Similar to the results using score-based fusion, the DT-based combination of two methods of HOG and ULBP leads to a decrease in the performance, while combining ConvEllip and FD increases the accuracy by about 3%; however, with a bigger gap between sensitivity and specificity. Interestingly, the performance of DT based on three methods, including ULBP and HOG, is much better than the other two decision and score level fusions. The same methods (HOG and ULBP) involved in DT based on five methods increase the internal testing accuracy by 4%, while it stays almost the same on external testing.

Three methods of DT show marginal improvements in internal and external testing, while DT, based on five methods, gives maximum accuracy improvement of 6%, 2%, and 0% on internal, external, and external agreed testing sets.

## 7.3 Results Analysis

The performance of the top-performing fusion and DT-based methods is presented in the following tables 7.30 and 7.31 for comparison. Similar to what we did in chapters 5 and 6, the result boxes of the best-performing schemes are shaded in various colours depending on the rank of their top performances, with red, green, and blue designating the first, second, and third best-performing approaches (see Table 5.30 for colour ranking).

Overall, the score-based fusion using five classifiers performed the best on all testing sets using individual and agreed doctor's class labels. On the internal testing, the DT-based method with five classifiers and the score-based with three classifiers performed second and third best, respectively. The performance ranking is slightly different on the external testing; the decision-based fusion using five and three classifiers comes second on doctors A and C class labels, while the score-based scheme using two classifiers comes second on doctor B's class label. The least-performing scheme is a decision based on three classifiers on internal testing followed by DT with three classifiers.

Table 7.30: Fusion and DT result's comparison of internal and external testing.

| Best Fusion Scheme | Internal Dataset | | | External Dataset | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | Doctor A | | | Doctor B | | | Doctor C | | |
| | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Decision 3 Classifiers | 83 | 85 | 81 | 82 | 84 | 80 | 78 | 83 | 75 | 91 | 90 | 92 |
| Decision 5 Classifiers | 86 | 93 | 81 | 85 | 91 | 80 | 77 | 85 | 72 | 90 | 92 | 88 |
| Score 2 Classifiers | 86 | 81 | 91 | 81 | 80 | 82 | 79 | 80 | 78 | 86 | 82 | 90 |
| Score 3 Classifiers | 86 | 85 | 88 | 82 | 80 | 84 | 78 | 78 | 78 | 87 | 82 | 92 |
| Score 5 Classifiers | 90 | 91 | 89 | 87 | 89 | 86 | 81 | 85 | 78 | 92 | 90 | 94 |
| DT 2 Classifiers | 86 | 76 | 94 | 80 | 75 | 84 | 78 | 75 | 80 | 85 | 78 | 92 |
| DT 3 Classifiers | 84 | 81 | 86 | 83 | 82 | 84 | 79 | 80 | 78 | 89 | 90 | 88 |
| DT 5 Classifiers | 90 | 98 | 83 | 81 | 91 | 73 | 77 | 90 | 68 | 90 | 96 | 84 |

Table 7.31 presents the performance comparison on the external testing set using the agreed class labels. The score-based fusion of five classifiers performed the best, achieving maximum accuracy of 96% and a relatively small gap of 5% between sensitivity and specificity. Interestingly decision majority rule fusion based on both three and five classifiers ranked as the second and third best schemes, with five methods fusion giving 93% total accuracy and a 0% gap between sensitivity and specificity. The DT-based scheme using five classifiers performed the worse.

Table 7.31: Fusion and DT result's comparison of external testing using agreed ground truth.

| Method | Internal Dataset | | | External Dataset (DS2) Agreed Cases | | |
|---|---|---|---|---|---|---|
| | Acc | Reg | Irreg | Acc | Reg | Irreg |
| Decision 3 Classifiers | 83 | 81 | 83 | 93 | 90 | 95 |
| Decision 5 Classifiers | 86 | 93 | 81 | 93 | 93 | 93 |
| Score 2 Classifiers | 86 | 81 | 91 | 90 | 83 | 95 |
| Score 3 Classifiers | 84 | 80 | 88 | 90 | 80 | 98 |
| Score 5 Classifiers | 90 | 91 | 89 | 96 | 93 | 98 |
| DT 2 Classifiers | 86 | 76 | 94 | 89 | 77 | 98 |
| DT 3 Classifiers | 84 | 81 | 86 | 92 | 87 | 95 |
| DT 5 Classifiers | 90 | 98 | 83 | 92 | 97 | 88 |

Comparing these results with the best results achieved in chapter 5 using the methods based on the interpolated borderline analysis presented in Tables 5.31 and 5.32, we can observe improvements of 4% on the internal testing set. The improvements on the external dataset were 5%, 2%, 4%, and 4% for doctors A, B, and C, and their agreed class labels, respectively. This means the various multi-classifier schemes outperformed our best-performing methods by around 5%.

## 7.4 Summary and Conclusion

In this chapter, we attempted to complement the investigation of the work in chapters 5 and 6 by building multi-classification schemes using several mixes of schemes from the two different approaches. We attempted two types of fusion schemes as well as building DT as a hierarchical multi-classifier. The relatively limited amounts of experimental work confirmed the viability of using the three approaches to achieve considerable accuracy. In particular, when tested on the subset of the external dataset where all clinicians reached a unanimous decision, the various DT and all fused schemes achieved higher accuracy over and above the best-performing single participating schemes. The results of the best-performing multi-classifier schemes demonstrate that the proposed methods from the two different approaches of borderline and texture analysis complement each other. However, the HOG method led to either lower performances or did not improve accuracy. ULBP method, in contrast, improved the performances when combined with other borderline analysis-based methods. Although combining the methods using DT did not perform the best, it could be beneficial in building a hierarchical system, especially at the testing stage, since not all the methods contribute to the final decision and consequently give faster testing times. Furthermore, the predicted scores can be used as decision confidence to build the hierarchical classifier, whereby the decisions with low confidence can go through multiple classifiers to get to a better decision.

In the next chapter, we conclude the thesis and present our future works.

# Chapter 8: Conclusion and Future Works

The research investigations reported in this thesis aimed to develop automatic machine learning algorithms for analysing images in terms of the appearance of certain types of abnormal image features or shapes. The appearance of such abnormalities in different image modalities is of significant importance in crucial computer vision applications in various fields of science, engineering, medicine, and art. We were concerned with image abnormalities, the appearance of which indicates potentially serious defects in the imaged objects/complexes and require timely actions. Two application areas of interest were identified and investigated: Inspecting cracks in building glass facades and concrete construction material using digital camera images and determining irregularity properties of tumour lesion borders from US scan images. In the first case appearance of cracks endanger people and infrastructures, while lesion border irregularity is one of the main signs the medical communities use for assessing the malignancy of the tumours.

The nature and reasons for abnormal shapes/features appearing in images of any modalities are often very specific to the applications, the capturing devices, and the objects/sceneries recorded. Abnormal features of interest may be associated with different image-disconnected visible structural objects (as is the case with cracked glass/concrete panels) spread out in the image in an unpredicted manner or related to one object of interest (as is the case of tumour border irregularity). Furthermore, the examined images may contain image objects that do not exhibit abnormalities. For example, shadows of external objects reflected on glass panels should not be mistaken for cracks, although they may exhibit similar structures but do not persist over time as cracks. This means that in the first case of cracks, the image foreground needs to be segmented before testing all foreground segments for abnormality properties, while in the second type of thyroid nodule border, only the closed curve bordering a "segmented" nodule needs to be determined/estimated and tested for irregularity. These differences in the nature of the image pre-processing tasks in the two chosen cases add to the challenges associated with differences between image modalities. Hence, our investigations need to design and apply different pre-processing procedures in the two cases. In both cases, the pre-processing step produces curve-like segments (with possible zigzagging). In addition, the geometric properties of the isolated connected segments or the lesion border provide some common, but not identical, strategies for relevant feature extraction to determine the sought-after abnormalities.

Since our investigations in both cases are essentially related to image analysis, we found it natural to expand our investigations by utilising common texture analysis methods developed over the decades for machine learning in computer vision. We mainly adopted handcrafted feature models for this approach but conducted limited experiments using CNN models only to test their viabilities. For nodule border irregularity, the black-box nature of CNN decisions makes them less attractive to clinicians.

The extracted feature vectors, in both cases, were to be trained and tested by various commonly known classifiers. However, for this step, we faced challenges of different nature with regard to the availability of an adequate experimental dataset. In the case of detecting glass façade cracked panels, the restricting regulation on using UAVs made it challenging to obtain a sufficiently large dataset; we only managed to collect a limited amount of video recordings. In the case of lesion border irregularity, experimental datasets of US tumour scan images need to be collected by clinical centres according to ethical agreements. Although plenty of such images may have been recorded for some time, no ethical approval can be obtained for such historically recorded images, and they are not readily usable for training AI algorithms. We adopted a "reductionist" approach to overcome these challenges described below.

## 8.1 Crack Recognition in Building Material Case study

This case study originated with a project to develop an AI method to detect cracks in the high rising building using natural video images captured by commercially available cameras mounted on UAVs. We conducted a pilot project to build a prototype of a video recording system and developed initial AI methods for glass crack recognition. This was tested in a small area in China with promising results, and the pilot resulted in a reasonably sized dataset of videos. To overcome the challenge of developing well-performing glass façade panel crack recognition schemes with limited dataset size, we exploited the fact that the glass panels are significantly large and partitioned each panel into blocks of reasonable size and considered each as a separate glass image sample. Other publicly available concrete image datasets were also used in this case study. We evaluated our automatic crack recognition methods on glass façade and concrete surface visual images as two construction materials. Several glass datasets are collected and annotated using drones and Google downloads.

The initial task was focused on extracting suspect image objects/features that exhibit the characteristics of cracks. For that, we applied a pre-processing method consisting of edge/arc detection algorithms to obtain a set of connected sequences of adjacent pixels forming visible image features to be examined for abnormality indicators. The edge drawing (ED) algorithm was the most suitable edge detector due to its ability to extract thin edges/arcs. The extracted connected objects are mostly continuous but non-smooth open arcs of different sizes, but occasionally, we get small close arcs. Realising that geometric parameters computed from these objects encapsulated reliable information about their abnormalities, we developed feature vector representations of these parameters computed on the aggregate of these objects as input to the sought-after machine learning classification schemes. The feature vectors represent changes to curvature or connected pixel configurations along the extracted arcs beside the new innovative histogram of linearity (HOL). These proposed methods, trained and tested with the various experimental datasets, have demonstrated high-performance levels for façade glass crack recognition. The desire to search for abnormality indicators from the entire image rather than those associated with extracted suspect arcs motivated the use of common texture features such as ULBP and

HOG as well as CNN models. The significantly high performance of these schemes demonstrated that abnormality indicators of glass façade cracks are manifested in the entire image rather than just the suspect objects. These results also have shown the benefits of using non-traditional machine learning (CNN) based algorithms for this task. The achieved success justifies the extension of this work to detect faults in other building materials. Indeed, we developed and tested the same approaches for automatically detecting cracks in concrete with equally high performance. The proposed methods achieved accuracies between 70% using linearity on the low-resolution dataset of the glasses (DHD) and 99% using the CNN model on the publicly available concrete dataset. The performance of the methods based on HOL, connected pixel configurations, HOG, and ULBP has been improved by partitioning the input image into equal blocks to increase the feature vector dimensionality achieving 79%, 78%, 88%, and 91%, respectively, on the low-resolution DHD glass dataset. Partition-based ULBP achieved the highest overall accuracy among all handcrafted features at 95% and 96% when tested on the high-resolution glass dataset (D4K) and the concrete dataset, respectively.

A prototype of the crack recognition methods used in the glass part of this case study shows the effectiveness of the suggested methods. The handcrafted techniques are incorporated into prototype software that can recognise a crack in a video of glass panels and identifies the cracked video frame. The contributions of the thesis in this case study include collecting various datasets of the glass cracks, using some of the existing texture features, and developing several new handcrafted features and deep learning models for crack recognition achieving high performances. However, the proposed methods could not be evaluated on more generic crack datasets due to the lack of publicly available datasets on glass cracks and the difficulty of collecting new ones due to privacy limitations. Further, the presence of many reflections on the glass surfaces, and stains, paints, and holes on the concrete surfaces may have affected the performance of our proposed vision-based cracks recognition algorithms which are not addressed in this work. In addition, our methods do not assess individual cracks to show their types or severity in terms of size, massiveness, and width.

Finally, it is worth pointing out that our use of CNN models came later in this case study when these models started to make their mark on the machine learning scene and became the preferred state-of-the-art method in many computer vision application areas. Accordingly, the experimental results achieved with limited CNN models should only be considered as an attempt to benchmark the performance of our developed handcraft feature-based schemes against CNN.

## 8.2 Lesion Border Irregularity Recognition in US Nodule Scan Images

The objective of this case study of automatic border irregularity assessment of thyroid cancer nodules and the challenges faced are different to those encountered in the first case study. The non-availability of a sufficiently large dataset of US images suitable for immediate use for developing AI algorithms

was dealt with by recording a new dataset. Unlike the first case study, each US image contains only one region of interest representing a nodule. Rather than a fully automatic/manual border segmentation, a special pre-processing method was designed to estimate the lesion border via bi-cubic interpolation based on a small set of ROI points marked by experienced clinical radiographers. The more serious challenge facing the use of AI in this specific case (as in many medical applications) is the interobserver variability of classifying irregularity results in the absence of unique gold-standard ground truth. In the first case study, interobserver variability in determining if a glass panel is cracked or not is negligible, if any.

Similar to the first case study, we followed two approaches: (1) building AI schemes by analysing the geometric parameters extracted from the interpolated nodule border and (2) building texture analysis AI schemes extracted from pixel intensities in the immediate surrounding ribbon of the interpolated borderline. The first approach was presented in chapter 5, whereby several methods are proposed that assess the border irregularity using borderline distances to several reference shapes fitted to the border of the nodule. Such reference shapes include a fitted ellipse, a Gaussian shape, a convex hull, and a fitted ellipse from the convex hull. The distances are either used as a feature vector for input to a classifier, to build a distance function for further assessments in the frequency domain (FFT), or to be analysed by TDA for irregularity recognition. The distance function is used additionally for irregularity region visualisation. Another method not based on distances is inspired by FD and uses an irregularity index calculated from measurements of the perimeter of the border and fitted ellipse at different scales. These experimental results demonstrated beyond doubt the viability of determining irregularity of nodule border from US scan images using a small set of ROI points instead of manual/automatic segmentation.

The second approach includes several traditional ML and CNN-based methods for irregularity recognition utilising texture variations around the borderline. This was motivated by the success of the texture analysis methods developed for the building material cracks, and it is hoped that it indirectly reduces the effect of errors in marking the ROI points and the subsequent error of estimating the lesion border by interpolation. Vectorised texture features deployed for this part of the investigations, conducted in chapter 6, included ULBP, HOG and HOL. A ribbon of variable width is constructed based on either radial distance from the interpolated border or morphological erosion and dilations operations to restrict the texture analysis around the borderline for the various texture features. To further improve the performance of the methods, the ribbon was divided into several sectors to extract localized border irregularity features and increase the dimensionality of the feature vector. The HOG and ULBP vectorised features are extracted from the sectors, and the feature histograms are concatenated to build the feature vector for input to a classifier. Several attempts to improve the performance of the ULBP-based method have been undertaken, including (1) training a model with copies of the same image using a generic range of ribbon widths and (2) using lesion size category-

dependent ribbon widths as input to design a lesion-size adaptive scheme. Though the experimental results were not as good as those achieved based on the border interpolated curve, confirming the viability of using the various texture analysis of lesion border ribbons for recognising border irregularity. The last method of analysing the textures around the borderline is the CNN models based on VGG16 and ResNet50 architectures. The same ribbon approach is used to crop the regions surrounding the borderline for input into CNN models.

Generally, the methods of the first approach performed significantly better than the methods of the second. The border distance-based method achieved the highest accuracy of 87% when analysed using TDA. However, the FD-inspired method performed better, with a 0% gap between sensitivity and specificity and an accuracy of 86% on the internal testing set when it was evaluated using the proposed simple iterative classifier. The two methods achieved the same highest accuracy on external testing using doctor A ground truth. FD-inspired method on doctors A and B and fitted ellipse from the convex hull on doctor C's class labels were the next-best performing methods on the external testing set with fitted ellipse from the convex method reaching 88% accuracy and 0% gap between the sensitivity and specificity. Both FD-inspired and fitted ellipses from convex hull-based methods achieved the same highest performances of 92% and 2% gap between sensitivity and specificity on the external subset with agreed doctors' class labels. Overall, the FD-inspired method performed the best, followed by the vectorised distance function-based methods using a fitted ellipse of the convex hull and TDA. The distance methods using centroid and fitted ellipse as reference performed worse among all methods. CNN models were used to obtain slightly better performance than ROI-based approaches during internal testing; however, this performance decreased on external testing. The highest CNN performance was 89%, 84%, and 88% on internal, external (doctor C ground truth), and external agreed cases, respectively. However, the performances were still lower than the fusion-based methods.

The efficacy of the proposed methods is demonstrated by developing visualisation tools to illustrate the various methods developed for lesion border irregularity recognition. Given that various methods of the two approaches for irregularity recognition were most promising, if not of high performance, several fusion schemes of the methods are attempted to improve the proposed methods' performance further. The fusions were conducted at the decision and score levels, and a decision tree was used to build a hierarchical classifier by combining the method's decisions. All adopted fusion schemes achieved performance improvement of different significance over the constituent schemes. The score level fusion performed the best when probability predictions of 5 methods were fused, achieving the highest overall accuracy of 90% and a gap between sensitivity and specificity of 2% when tested on the internal dataset. The method's best accuracy on the external dataset is 92%, with a gap between sensitivity and specificity of 4% when doctor C's labelling is used. The highest overall accuracy of 96% is achieved on the external agreed cases between the three radiologists. In general, the score-based fusion performed the best, followed by combining the decisions using a decision tree, and the least performance improvement was

achieved by majority rule decision fusion. The performance improvements of the various fusion schemes indicate the complementary nature of our various features extracted for border irregularity recognition.

In conclusion, the high performance of the various methods proposed in this part of the work shows the efficacy of the methods based on the ROI points marked by the doctors for border irregularity recognitions without the need for manual/automatic segmentations. The work also showed that combining some of the methods from both approaches of ROI only and pixel intensity analysis can further improve performance.

## 8.3 Future Work

Here, we present a list of future work relating to the objectives of this thesis for both case studies of crack recognition in building materials and border irregularity recognition of thyroid nodules. These suggested additional research investigations go beyond the obvious need for ongoing finetuning of the developed schemes for improved performance or widening their testing on new and larger datasets.

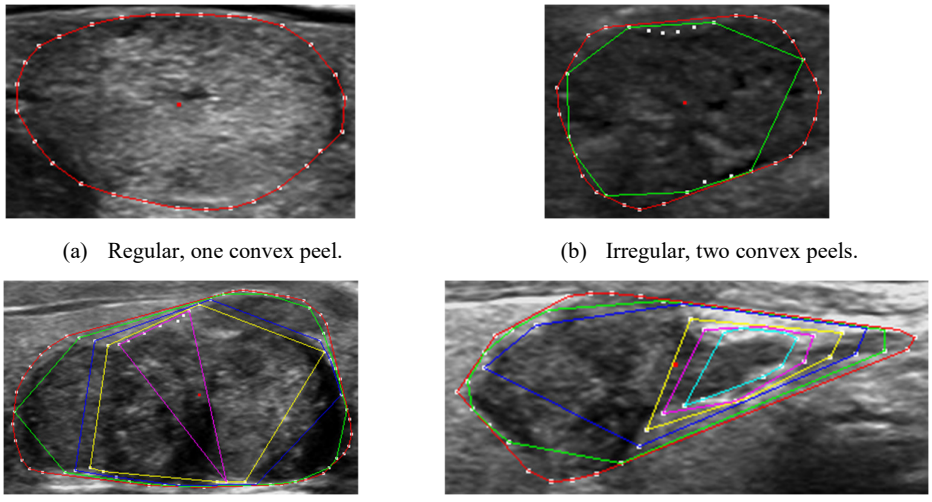
1. Further development of our work on glass panel cracks recognition to determine crack severity, depth of cracks and growth of cracks over time. Extend the work on glass façade inspection beyond the glass panels to inspect the metal and sealant materials surrounding the glass panels for serious building faults. Some of these tasks may require imaging sensors other than digital video cameras. Integrating these new methods with the existing schemes for the glass and concrete material crack recognition schemes to develop an entire 3D system to be used for regular assessment of entire building safety.

2. Our list of glass/concrete crack recognition schemes is to be tested and possibly modified or expanded by using other features to detect faults in other building constructs/complexes, such as recognising faulty solar panels in solar energy-generating farms.

3. Extend the current work on nodule border irregularity to develop schemes to separately assess the so-called structural and local irregularities. The structural border irregularity represents large protrusions and indentations of the borderline, which can also be referred to as shape irregularity since it represents overall nodule irregularity. On the other hand, local irregularity is represented in the small zigzags along the borderline. These two types of irregularity might have different meanings and weights for assessing the malignancy of the lesion [164], [182] and, therefore, worth assessing independently.

The two types of border irregularities can be captured by analysing the distance function based on any of the proposed methods in sections 5.4.3 to 5.4.8 in the frequency domain. It is natural to assume that the structural irregularities are represented in the lower and the local irregularities in the higher frequency ranges of the FFT spectrum. Consequently, the FFT-based method proposed in section 5.4.9.1 can be extended so that we extract two irregularity indexes or feature vectors from the upper and lower frequency band of the FFT spectrum. Finding the optimal threshold between the two frequency bands is not trivial and might need to be determined empirically, and it might be avoided using low and high-pass frequency filters.

4. Convex peel numbers and their centroids can be exploited for shape complexity analysis of cancer lesions. In chapter 5, we used the convex hull of the RIO border points and its fitting ellipse for extracting distance functions. The position of the ROI points inside the convex hull surely impacts the nodule shape and its irregularity characteristics. One can iteratively generate a sequence of convex hulls, called ***convex peels***, by removing the current convex hull corners from the set ROI points, starting with the initial convex hull, and a new convex hull is constructed until all ROI points are used. Figure 8.1 below illustrates different convex peels of samples of our dataset, including regular and irregular shapes. The nodule's shape in Figure (a) is almost an ellipse and yields one convex peel, while the nodule in Figure (b) has a more irregular border and yields two convex peels. The case with a more complex shape in Figure (c) yields five convex peels, while the most complex shape with elongation on the right side shown in Figure (d) yields six convex peels.



(a) Regular, one convex peel. (b) Irregular, two convex peels.

(c) Irregular, five convex peels. (d) Irregular, six convex peels.

Figure 8.1: Exploiting the number of convex peels and their centroids for lesion shape complexity analysis. White dots are the ROI point; different coloured curves are the peel starting with red for the outer one.

The number of convex peels of a shape could give indications about the complexity of that shape. Generally, the more convex peels a shape has, the more complex or irregular the shape can be. We intend to investigate the effect of the number of peels, the trace of their centroids and the use of distance functions defined by these peels to improve irregularity recognition.

# References


[1] “Look Out Below: Glass Panel Falls From 27th Floor.” Accessed: Sep. 28, 2022. [Online]. Available: https://www.nytimes.com/2008/04/02/nyregion/02glass.html

[2] D. Hongbo, “Courtesy Hongbo Du.” Sep. 10, 2022.

[3] “Video crack in the glass.” Accessed: Sep. 28, 2022. [Online]. Available: https://www.shutterstock.com/video/clip-11638904-video-crack-glass

[4] “Self-Healing Concrete Could Rid Sidewalks Of Cracking.” Accessed: Sep. 28, 2022. [Online]. Available: https://www.businessinsider.com/bacteria-make-concrete-self-healing-2012-11?r=US&IR=T

[5] R. C. Gonzalez and R. E. Woods, *Digital image processing*, Fourth edition, Global edition. New York, NY: Pearson, 2018.

[6] S. Belongie, J. Malik, and J. Puzicha, “Shape matching and object recognition using shape contexts,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002, doi: 10.1109/34.993558.

[7] D. N. Parmar and B. B. Mehta, “Face Recognition Methods & Applications,” *arXiv:1403.0485 [cs]*, Mar. 2014, Accessed: Mar. 31, 2022. [Online]. Available: http://arxiv.org/abs/1403.0485

[8] P. Zhu *et al.*, “Detection and Tracking Meet Drones Challenge,” *arXiv:2001.06303 [cs]*, Oct. 2021, Accessed: Mar. 31, 2022. [Online]. Available: http://arxiv.org/abs/2001.06303

[9] D. Avola, L. Cinque, A. Fagioli, G. Foresti, and A. Mecca, “Ultrasound Medical Imaging Techniques: A Survey,” *ACM Comput. Surv.*, vol. 54, no. 3, pp. 1–38, Apr. 2022, doi: 10.1145/3447243.

[10] J. Ding, H. Cheng, C. Ning, J. Huang, and Y. Zhang, “Quantitative Measurement for Thyroid Cancer Characterization Based on Elastography,” *Journal of Ultrasound in Medicine*, vol. 30, no. 9, pp. 1259–1266, Sep. 2011, doi: 10.7863/jum.2011.30.9.1259.

[11] U. Acharya *et al.*, “Evolutionary Algorithm-Based Classifier Parameter Tuning for Automatic Ovarian Cancer Tissue Characterization and Classification,” *Ultraschall in Med*, vol. 35, no. 03, pp. 237–245, Dec. 2012, doi: 10.1055/s-0032-1330336.

[12] J. C. Gomes *et al.*, “IKONOS: an intelligent tool to support diagnosis of COVID-19 by texture analysis of X-ray images,” *Res. Biomed. Eng.*, Sep. 2020, doi: 10.1007/s42600-020-00091-7.

[13] R. M. Haralick, K. Shanmugam, and I. Dinstein, “Textural Features for Image Classification,” *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973, doi: 10.1109/TSMC.1973.4309314.

[14] M. Liu, Y. He, and B. Ye, “Image Zernike moments shape feature evaluation based on image reconstruction,” *Geo-spatial Information Science*, vol. 10, no. 3, pp. 191–195, Jan. 2007, doi: 10.1007/s11806-007-0060-x.

[15] U. R. Acharya, O. Faust, S. V. Sree, F. Molinari, and J. S. Suri, “ThyroScreen system: High resolution ultrasound thyroid image characterization into benign and malignant classes using novel combination of texture and discrete wavelet transform,” *Computer Methods and Programs in Biomedicine*, vol. 107, no. 2, pp. 233–241, Aug. 2012, doi: 10.1016/j.cmpb.2011.10.001.

[16] W.-J. Wu and W. K. Moon, “Ultrasound Breast Tumor Image Computer-Aided Diagnosis With Texture and Morphological Features,” *Academic Radiology*, vol. 15, no. 7, pp. 873–880, Jul. 2008, doi: 10.1016/j.acra.2008.01.010.

[17] U. R. Acharya, S. V. Sree, L. Saba, F. Molinari, S. Guerriero, and J. S. Suri, “Ovarian tumor characterization and classification using ultrasound-a new online paradigm,” *J Digit Imaging*, vol. 26, no. 3, pp. 544–553, Jun. 2013, doi: 10.1007/s10278-012-9553-8.

[18] P. Shanmugavadivu and V. Sivakumar, “Fractal Dimension Based Texture Analysis of Digital Images,” *Procedia Engineering*, vol. 38, pp. 2981–2986, 2012, doi: 10.1016/j.proeng.2012.06.348.

[19] S. Khazendar *et al.*, “Automated characterisation of ultrasound images of ovarian tumours: the diagnostic accuracy of a support vector machine and image processing with a local binary pattern operator,” *Facts Views Vis Obgyn*, vol. 7, no. 1, pp. 7–15, 2015.

[20] T. Napoléon and A. Alfalou, “Pose invariant face recognition: 3D model from single photo,” *Optics and Lasers in Engineering*, vol. 89, pp. 150–161, Feb. 2017, doi: 10.1016/j.optlaseng.2016.06.019.

[21] T. Ojala, M. Pietikäinen, and D. Harwood, “A comparative study of texture measures with classification based on featured distributions,” *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, Jan. 1996, doi: 10.1016/0031-3203(95)00067-4.

[22] T. Tuncer, S. Dogan, and F. Ozyurt, “An automated Residual Exemplar Local Binary Pattern and iterative ReliefF based COVID-19 detection method using chest X-ray image,” *Chemometrics and Intelligent Laboratory Systems*, vol. 203, p. 104054, Aug. 2020, doi: 10.1016/j.chemolab.2020.104054.

[23] H. Tan, B. Yang, and Z. Ma, “Face recognition based on the fusion of global and local HOG features of face images,” *IET Computer Vision*, vol. 8, no. 3, pp. 224–234, Jun. 2014, doi: 10.1049/iet-cvi.2012.0302.

[24] L. Meng, Z. Wang, Y. Fujikawa, and S. Oyanagi, “Detecting cracks on a concrete surface using histogram of oriented gradients,” in *2015 International Conference on Advanced Mechatronic Systems (ICAMechS)*, Aug. 2015, pp. 103–107. doi: 10.1109/ICAMechS.2015.7287137.

[25] V. A, D. Hebbar, V. S. Shekhar, K. N. B. Murthy, and S. Natarajan, “Two Novel Detector-Descriptor Based Approaches for Face Recognition Using SIFT and SURF,” *Procedia Computer Science*, vol. 70, pp. 185–197, 2015, doi: 10.1016/j.procs.2015.10.070.

[26] M. Alam, D. Thapa, J. I. Lim, D. Cao, and X. Yao, “Quantitative characteristics of sickle cell retinopathy in optical coherence tomography angiography,” *Biomed. Opt. Express*, vol. 8, no. 3, p. 1741, Mar. 2017, doi: 10.1364/BOE.8.001741.

[27] Ming Liang and Xiaolin Hu, “Recurrent convolutional neural network for object recognition,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 3367–3375. doi: 10.1109/CVPR.2015.7298958.

[28] S. Srivastava, A. V. Divekar, C. Anilkumar, I. Naik, V. Kulkarni, and V. Pattabiraman, “Comparative analysis of deep learning image detection algorithms,” *J Big Data*, vol. 8, no. 1, p. 66, Dec. 2021, doi: 10.1186/s40537-021-00434-w.

[29] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen, “Medical image classification with convolutional neural network,” in *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*, Singapore, Dec. 2014, pp. 844–848. doi: 10.1109/ICARCV.2014.7064414.

[30] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, “Image Segmentation Using Deep Learning: A Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2021, doi: 10.1109/TPAMI.2021.3059968.

[31] A. Torfi, R. A. Shirvani, Y. Keneshloo, N. Tavaf, and E. A. Fox, “Natural Language Processing Advancements By Deep Learning: A Survey,” 2020, doi: 10.48550/ARXIV.2003.01200.

[32] Y. Chauvin and D. E. Rumelhart, Eds., *Backpropagation: theory, architectures, and applications*. Hillsdale, N.J: Lawrence Erlbaum Associates, 1995.

[33] *CNN Architecture figure*. [Online]. Available: https://discuss.boardinfinity.com/t/what-do-you-mean-by-convolutional-neural-network/8533

[34] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J Big Data*, vol. 6, no. 1, p. 60, Dec. 2019, doi: 10.1186/s40537-019-0197-0.

[35] Y. Gao and K. M. Mosalam, "Deep Transfer Learning for Image-Based Structural Damage Recognition: Deep transfer learning for image-based structural damage recognition," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 9, pp. 748–768, Sep. 2018, doi: 10.1111/mice.12363.

[36] F. Mohammad, A. AlZoubi, H. Du, and S. Jassim, "A generic approach for automatic crack recognition in buildings glass facade and concrete structures," in *Thirteenth International Conference on Digital Image Processing (ICDIP 2021)*, Singapore, Singapore, Jun. 2021, p. 70. doi: 10.1117/12.2601061.

[37] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, Jun. 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848.

[38] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," *arXiv:1411.1792 [cs]*, Nov. 2014, Accessed: Apr. 10, 2022. [Online]. Available: http://arxiv.org/abs/1411.1792

[39] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps," *arXiv:1312.6034 [cs]*, Apr. 2014, Accessed: Apr. 10, 2022. [Online]. Available: http://arxiv.org/abs/1312.6034

[40] Z. Qin, F. Yu, C. Liu, X. Chen, ,George Mason University, 4400 University Dr, Fairfax, VA 22030, USA, and ,Clarkson University, 8 Clarkson Ave, Potsdam, NY 13699, USA, "How convolutional neural networks see the world --- A survey of convolutional neural network visualization methods," *Mathematical Foundations of Computing*, vol. 1, no. 2, pp. 149–180, 2018, doi: 10.3934/mfc.2018008.

[41] N. K. Mahobia, "Mahobia, N.K., Patel, R.D., Sheikh, N.W., Singh, S.K., Mishra, A. and Dhardubey, R., 2010. Validation method used in quantitative structure activity relationship. Der Pharma Chemica, 2(5), pp.260-271.," *Der Pharma Chemica*, vol. 2(5), p. pp.260-271..

[42] T.-T. Wong, “Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation,” *Pattern Recognition*, vol. 48, no. 9, pp. 2839–2846, Sep. 2015, doi: 10.1016/j.patcog.2015.03.009.

[43] J. Shao, “Linear Model Selection by Cross-validation,” *Journal of the American Statistical Association*, vol. 88, no. 422, pp. 486–494, Jun. 1993, doi: 10.1080/01621459.1993.10476299.

[44] A. Ericsson and J. Karlsson, “Measures for Benchmarking of Automatic Correspondence Algorithms,” *J Math Imaging Vis*, vol. 28, no. 3, pp. 225–241, Oct. 2007, doi: 10.1007/s10851-007-0018-5.

[45] “ROC Curve.” Apr. 10, 2022. [Online]. Available: https://towardsdatascience.com/a-quick-guide-to-auc-roc-in-machine-learning-models-f0aedb78fbad

[46] F. Faschingbauer *et al.*, “Automatic Texture-Based Analysis in Ultrasound Imaging of Ovarian Masses,” *Ultraschall in Med*, vol. 34, no. 02, pp. 145–150, May 2012, doi: 10.1055/s-0031-1299331.

[47] F. Anowar, S. Sadaoui, and B. Selim, “Conceptual and empirical comparison of dimensionality reduction algorithms (PCA, KPCA, LDA, MDS, SVD, LLE, ISOMAP, LE, ICA, t-SNE),” *Computer Science Review*, vol. 40, p. 100378, May 2021, doi: 10.1016/j.cosrev.2021.100378.

[48] V. Kamble and K. M. Bhurchandi, “No-reference image quality assessment algorithms: A survey,” *Optik*, vol. 126, no. 11–12, pp. 1090–1097, Jun. 2015, doi: 10.1016/j.ijleo.2015.02.093.

[49] N. Kanopoulos, N. Vasanthavada, and R. L. Baker, “Design of an image edge detection filter using the Sobel operator,” *IEEE J. Solid-State Circuits*, vol. 23, no. 2, pp. 358–367, Apr. 1988, doi: 10.1109/4.996.

[50] S. Bhairannawar, A. Patil, A. Janmane, and M. Huilgol, “Color image enhancement using Laplacian filter and contrast limited adaptive histogram equalization,” in *2017 Innovations in Power and Advanced Computing Technologies (i-PACT)*, Vellore, Apr. 2017, pp. 1–5. doi: 10.1109/IPACT.2017.8244991.

[51] A. J. Abboud, “Quality Aware Adaptive Biometric Systems,” The University of Buckingham, 2011.

[52] X.-Y. Zhang, L. Ge, and T.-F. Wang, “Entropy-Based Local Histogram Equalization for Medical Ultrasound Image Enhancement,” in *2008 2nd International Conference on Bioinformatics and Biomedical Engineering*, Shanghai, China, May 2008, pp. 2427–2429. doi: 10.1109/ICBBE.2008.939.

[53] Surbhi and V. Arora, “ROI Segmentation for Feature Extraction from Human Facial Images,” 2012, doi: 10.48550/ARXIV.1207.2922.

[54] J. Dabass, S. Arora, R. Vig, and M. Hanmandlu, “Segmentation Techniques for Breast Cancer Imaging Modalities-A Review,” in *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, Noida, India, Jan. 2019, pp. 658–663. doi: 10.1109/CONFLUENCE.2019.8776937.

[55] hafsa ouchra and abdessamad belangour, “Object detection approaches in images: a survey,” in *Thirteenth International Conference on Digital Image Processing (ICDIP 2021)*, Singapore, Singapore, Jun. 2021, p. 85. doi: 10.1117/12.2601452.

[56] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation.” arXiv, Oct. 22, 2014. Accessed: Oct. 01, 2022. [Online]. Available: http://arxiv.org/abs/1311.2524

[57] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” 2015, doi: 10.48550/ARXIV.1506.02640.

[58] B. Wu, A. Wan, F. Iandola, P. H. Jin, and K. Keutzer, “SqueezeDet: Unified, Small, Low Power Fully Convolutional Neural Networks for Real-Time Object Detection for Autonomous Driving,” 2016, doi: 10.48550/ARXIV.1612.01051.

[59] R. A. Kirsch, “Computer determination of the constituent structure of biological images,” *Computers and Biomedical Research*, vol. 4, no. 3, pp. 315–328, Jun. 1971, doi: 10.1016/0010-4809(71)90034-6.

[60] G. S. Robinson, “Edge detection by compass gradient masks,” *Computer Graphics and Image Processing*, vol. 6, no. 5, pp. 492–501, Oct. 1977, doi: 10.1016/S0146-664X(77)80024-5.

[61] L. G. Roberts, “Machine perception of 3-D solids-series,” *Optical and Electro-Optical Information Processing MIT Press, Cambridge*, vol. 159–197, 1965.

[62] J. M. Prewitt, “Object enhancement and extraction,” *Picture processing and Psychopictorics*, vol. 15–19, no. 10.1, 1970.

[63] J. Canny, “A Computational Approach to Edge Detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, Art. no. 6, Nov. 1986, doi: 10.1109/TPAMI.1986.4767851.

[64] C. Topal and C. Akinlar, “Edge Drawing: A combined real-time edge and segment detector,” *Journal of Visual Communication and Image Representation*, vol. 23, no. 6, pp. 862–872, Aug. 2012, doi: 10.1016/j.jvcir.2012.05.004.

[65] L. Liu, J. Chen, P. Fieguth, G. Zhao, R. Chellappa, and M. Pietikäinen, “From BoW to CNN: Two Decades of Texture Representation for Texture Classification,” *Int J Comput Vis*, vol. 127, no. 1, pp. 74–109, Jan. 2019, doi: 10.1007/s11263-018-1125-z.

[66] K. Mikolajczyk and C. Schmid, “An Affine Invariant Interest Point Detector,” in *Computer Vision — ECCV 2002*, vol. 2350, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 128–142. doi: 10.1007/3-540-47969-4_9.

[67] J. G�rding and T. Lindeberg, “Direct computation of shape cues using scale-adapted spatial derivative operators,” *Int J Comput Vision*, vol. 17, no. 2, pp. 163–191, Feb. 1996, doi: 10.1007/BF00058750.

[68] Tai Sing Lee, “Image representation using 2D Gabor wavelets,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 10, pp. 959–971, Oct. 1996, doi: 10.1109/34.541406.

[69] T. Leung and J. Malik, “[No title found],” *International Journal of Computer Vision*, vol. 43, no. 1, pp. 29–44, 2001, doi: 10.1023/A:1011126920638.

[70] C. Schmid, “Constructing models for content-based image retrieval,” in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, Kauai, HI, USA, 2001, vol. 2, p. II-39-II–45. doi: 10.1109/CVPR.2001.990922.

[71] M. Crosier and L. D. Griffin, “Using Basic Image Features for Texture Classification,” *Int J Comput Vis*, vol. 88, no. 3, pp. 447–460, Jul. 2010, doi: 10.1007/s11263-009-0315-0.

[72] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, San Diego, CA, USA, 2005, vol. 1, pp. 886–893. doi: 10.1109/CVPR.2005.177.

[73] U. Kandaswamy, S. A. Schuckers, and D. Adjeroh, “Comparison of Texture Analysis Schemes Under Nonideal Conditions,” *IEEE Trans. on Image Process.*, vol. 20, no. 8, pp. 2260–2275, Aug. 2011, doi: 10.1109/TIP.2010.2101612.

[74] R. Bajcsy, “Computer identification of visual surfaces,” *Computer Graphics and Image Processing*, vol. 2, no. 2, pp. 118–130, Oct. 1973, doi: 10.1016/0146-664X(73)90023-3.

[75] B. Mandelbrot, “How Long Is the Coast of Britain? Statistical Self-Similarity and Fractional Dimension,” *Science*, vol. 156, no. 3775, pp. 636–638, May 1967, doi: 10.1126/science.156.3775.636.

[76] Y. Xu, H. Ji, and C. Fermüller, “Viewpoint Invariant Texture Description Using Fractal Analysis,” *Int J Comput Vis*, vol. 83, no. 1, pp. 85–100, Jun. 2009, doi: 10.1007/s11263-009-0220-6.

[77] M. Tuceryan and A. K. Jain, “TEXTURE ANALYSIS,” in *Handbook of Pattern Recognition and Computer Vision*, WORLD SCIENTIFIC, 1993, pp. 235–276. doi: 10.1142/9789814343138_0010.

[78] J. Zhang and T. Tan, “Brief review of invariant texture analysis methods,” *Pattern Recognition*, vol. 35, no. 3, pp. 735–747, Mar. 2002, doi: 10.1016/S0031-3203(01)00074-7.

[79] S. Baheerathan, F. Albregtsen, and H. E. Danielsen, “New texture features based on the complexity curve,” *Pattern Recognition*, vol. 32, no. 4, pp. 605–618, Apr. 1999, doi: 10.1016/S0031-3203(98)00122-8.

[80] R. K. Goyal, W. L. Goh, D. P. Mital, and K. L. Chan, “Scale and rotation invariant texture analysis based on structural property,” in *Proceedings of IECON ’95 - 21st Annual Conference on IEEE Industrial Electronics*, Orlando, FL, USA, 1995, vol. 2, pp. 1290–1294. doi: 10.1109/IECON.1995.483983.

[81] G. Eichmann and T. Kasparis, “Topologically invariant texture descriptors,” *Computer Vision, Graphics, and Image Processing*, vol. 41, no. 3, pp. 267–281, Mar. 1988, doi: 10.1016/0734-189X(88)90102-8.

[82] R. Chellappa and S. Chatterjee, “Classification of textures using Gaussian Markov random fields,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 4, pp. 959–963, Aug. 1985, doi: 10.1109/TASSP.1985.1164641.

[83] A. Khotanzad and R. L. Kashyap, “Feature selection for texture recognition based on image synthesis,” *IEEE Trans. Syst., Man, Cybern.*, vol. 17, no. 6, pp. 1087–1095, Nov. 1987, doi: 10.1109/TSMC.1987.6499322.

[84] B. Julesz, “Experiments in the Visual Perception of Texture,” *Sci Am*, vol. 232, no. 4, pp. 34–43, Apr. 1975, doi: 10.1038/scientificamerican0475-34.

[85] Ming-Kuei Hu, “Visual pattern recognition by moment invariants,” *IEEE Trans. Inform. Theory*, vol. 8, no. 2, pp. 179–187, Feb. 1962, doi: 10.1109/TIT.1962.1057692.

[86] M. Pietikäinen, T. Ojala, and Z. Xu, “Rotation-invariant texture classification using feature distributions,” *Pattern Recognition*, vol. 33, no. 1, pp. 43–52, Jan. 2000, doi: 10.1016/S0031-3203(99)00032-1.

[87] F. Tomita and S. Tsuji, *Computer Analysis of Visual Textures*. Boston, MA: Springer US, 1990. doi: 10.1007/978-1-4613-1553-7.

[88] H. R. Boveiri, “On pattern classification using statistical moments.,” *International Journal of Signal Processing, Image Processing and Pattern Recognition, 3(4)*, 2010.

[89] A. Materka, “Materka, A., & Strzelecki, M. (1998). Texture analysis methods–a review. Technical university of lodz, institute of electronics, COST B11 report, Brussels, 10(1.97), 4968.,” *echnical university of lodz, institute of electronics, COST B11 report, Brussels*, 1998, [Online]. Available: https://www.researchgate.net/profile/Andrzej-Materka/publication/249723259_Texture_Analysis_Methods_-_A_Review/links/02e7e51ef8d539a9da000000/Texture-Analysis-Methods-A-Review.pdf

[90] R. M. Haralick, “Statistical and structural approaches to texture,” *Proc. IEEE*, vol. 67, no. 5, pp. 786–804, 1979, doi: 10.1109/PROC.1979.11328.

[91] A. Rosenfeld and J. S. Weszka, “Picture Recognition,” in *Digital Pattern Recognition*, vol. 10, K. S. Fu, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1980, pp. 135–166. doi: 10.1007/978-3-642-67740-3_5.

[92] J. G. Daugman, “Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters,” *J. Opt. Soc. Am. A*, vol. 2, no. 7, p. 1160, Jul. 1985, doi: 10.1364/JOSAA.2.001160.

[93] S. G. Mallat, “Multifrequency channel decompositions of images and wavelet models,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 12, pp. 2091–2110, Dec. 1989, doi: 10.1109/29.45554.

[94] R. O. Duda and P. E. Hart, “Use of the Hough transformation to detect lines and curves in pictures,” *Commun. ACM*, vol. 15, no. 1, pp. 11–15, Jan. 1972, doi: 10.1145/361237.361242.

[95] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002, doi: 10.1109/TPAMI.2002.1017623.

[96] M. Pietikäinen and G. Zhao, “Two decades of local binary patterns,” in *Advances in Independent Component Analysis and Learning Machines*, Elsevier, 2015, pp. 175–210. doi: 10.1016/B978-0-12-802806-3.00009-9.

[97] Y. Liu, K. Xu, and J. Xu, “An Improved MB-LBP Defect Recognition Approach for the Surface of Steel Plates,” *Applied Sciences*, vol. 9, no. 20, p. 4222, Oct. 2019, doi: 10.3390/app9204222.

[98] F. Mohammad, A. AlZoubi, D. Hongbo, and S. Jassim, “Automatic glass crack recognition for high building façade inspection,” in *Mobile Multimedia/Image Processing, Security, and Applications 2020*, Online Only, United States, May 2020, p. 32. doi: 10.1117/12.2567409.

[99] K. Meena and A. Suruliandi, “Local binary patterns and its variants for face recognition,” in *2011 International Conference on Recent Trends in Information Technology (ICRTIT)*, Chennai, India, Jun. 2011, pp. 782–786. doi: 10.1109/ICRTIT.2011.5972286.

[100] D. Al-Karawi *et al.*, “OC04.04: A machine-learning algorithm to distinguish benign and malignant adnexal tumours from ultrasound images,” *Ultrasound Obstet Gynecol*, vol. 54, no. S1, pp. 9–10, Oct. 2019, doi: 10.1002/uog.20445.

[101] H. Zhou, R. Wang, and C. Wang, “A novel extended local-binary-pattern operator for texture analysis,” *Information Sciences*, vol. 178, no. 22, pp. 4314–4325, Nov. 2008, doi: 10.1016/j.ins.2008.07.015.

[102] M. Pietikäinen, A. Hadid, G. Zhao, and T. Ahonen, *Computer Vision Using Local Binary Patterns*, vol. 40. London: Springer London, 2011. doi: 10.1007/978-0-85729-748-8.

[103] N. Kouiroukidis and G. Evangelidis, “The Effects of Dimensionality Curse in High Dimensional kNN Search,” in *2011 15th Panhellenic Conference on Informatics*, Kastoria, Greece, Sep. 2011, pp. 41–45. doi: 10.1109/PCI.2011.45.

[104] D. Al-Karawi, “Texture Analysis based Machine Learning Algorithms for Ultrasound Ovarian Tumour Image Classification within Clinical Practices,” The University of Buckingham, The University of Buckingham, 2019.

[105] T. Ahonen, J. Matas, C. He, and M. Pietikäinen, “Rotation Invariant Image Description with Local Binary Pattern Histogram Fourier Features,” in *Image Analysis*, vol. 5575, A.-B. Salberg, J. Y. Hardeberg, and R. Jenssen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 61–70. doi: 10.1007/978-3-642-02230-2_7.

[106] T. Mäenpää and M. Pietikäinen, “Classification with color and texture: jointly or separately?,” *Pattern Recognition*, vol. 37, no. 8, pp. 1629–1640, Aug. 2004, doi: 10.1016/j.patcog.2003.11.011.

[107] Zhenhua Guo, Lei Zhang, and D. Zhang, “A Completed Modeling of Local Binary Pattern Operator for Texture Classification,” *IEEE Trans. on Image Process.*, vol. 19, no. 6, pp. 1657–1663, Jun. 2010, doi: 10.1109/TIP.2010.2044957.

[108] G. Zhao and M. Pietikainen, “Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 915–928, Jun. 2007, doi: 10.1109/TPAMI.2007.1110.

[109] Baochang Zhang, Yongsheng Gao, Sanqiang Zhao, and Jianzhuang Liu, “Local Derivative Pattern Versus Local Binary Pattern: Face Recognition With High-Order Local Pattern Descriptor,” *IEEE Trans. on Image Process.*, vol. 19, no. 2, pp. 533–544, Feb. 2010, doi: 10.1109/TIP.2009.2035882.

[110] Z. Guo, Q. Li, J. You, D. Zhang, and W. Liu, “Local directional derivative pattern for rotation invariant texture classification,” *Neural Comput & Applic*, vol. 21, no. 8, pp. 1893–1904, Nov. 2012, doi: 10.1007/s00521-011-0586-6.

[111] S. S. Teoh and T. Braunl, “Performance evaluation of HOG and Gabor features for vision-based vehicle detection,” in *2015 IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, Penang, Malaysia, Nov. 2015, pp. 66–71. doi: 10.1109/ICCSCE.2015.7482159.

[112] B. B. Mandelbrot and J. A. Wheeler, “*The Fractal Geometry of Nature*,” *American Journal of Physics*, vol. 51, no. 3, pp. 286–287, Mar. 1983, doi: 10.1119/1.13295.

[113] B. B. Mandelbrot, Dann. E. Passoja, and A. J. Paullay, “Fractal character of fracture surfaces of metals,” *Nature*, vol. 308, no. 5961, pp. 721–722, Apr. 1984, doi: 10.1038/308721a0.

[114] V. T. Y. Ng and T. K. Lee, “Measuring border irregularities of skin lesions using fractal dimensions,” Beijing, China, Sep. 1996, pp. 64–72. doi: 10.1117/12.253385.

[115] E. Aydındağ Bayrak and P. Kırcı, “The Application of Fractal Analysis on Thyroid Ultrasound Images,” *ACIN*, pp. 83–90, Dec. 2019, doi: 10.26650/acin.496129.

[116] S. R. Nayak, J. Mishra, and G. Palai, “Analysing roughness of surface through fractal dimension: A review,” *Image and Vision Computing*, vol. 89, pp. 21–34, Sep. 2019, doi: 10.1016/j.imavis.2019.06.015.

[117] I. Konatar, T. Popovic, and N. Popovic, “Box-Counting Method in Python for Fractal Analysis of Biomedical Images,” in *2020 24th International Conference on Information Technology (IT)*, Zabljak, Montenegro, Feb. 2020, pp. 1–4. doi: 10.1109/IT48810.2020.9070454.

[118] L. S. Liebovitch and T. Toth, “A fast algorithm to determine fractal dimensions by box counting,” *Physics Letters A*, vol. 141, no. 8–9, pp. 386–390, Nov. 1989, doi: 10.1016/0375-9601(89)90854-2.

[119] Y.-D. Zhang *et al.*, “Fractal Dimension Estimation for Developing Pathological Brain Detection System Based on Minkowski-Bouligand Method,” *IEEE Access*, vol. 4, pp. 5937–5947, 2016, doi: 10.1109/ACCESS.2016.2611530.

[120] M. K. Biswas, T. Ghose, S. Guha, and P. K. Biswas, “Fractal dimension estimation for texture images: A parallel approach,” *Pattern Recognition Letters*, vol. 19, no. 3–4, pp. 309–313, Mar. 1998, doi: 10.1016/S0167-8655(98)00002-6.

[121] J. Weston, A. Elisseeff, B. Schölkopf, and M. Tipping, “Use of the zero norm with linear models and kernel methods.,” *The Journal of Machine Learning Research, 3, 1439-1461*, 2003.

[122] N. Sánchez-Maroño, A. Alonso-Betanzos, and M. Tombilla-Sanromán, “Filter Methods for Feature Selection – A Comparative Study,” in *Intelligent Data Engineering and Automated Learning - IDEAL 2007*, vol. 4881, H. Yin, P. Tino, E. Corchado, W. Byrne, and X. Yao, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 178–187. doi: 10.1007/978-3-540-77226-2_19.

[123] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, “Gene Selection for Cancer Classification using Support Vector Machines,” *Machine Learning*, vol. 46, no. 1/3, pp. 389–422, 2002, doi: 10.1023/A:1012487302797.

[124] V. Fonti and E. Belitser, “Feature selection using lasso.,” *VU Amsterdam research paper in business analytics, 30, 1-25*, 2017.

[125] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[126] *Flickr Homepage/Shanghai tower*. 2022. Accessed: Sep. 18, 2022. [Online]. Available: https://www.flickr.com/photos/fesign/32093570130/in/photostream/lightbox/

[127] *Drone Facade Inspections Sydney Homepage*. 2022. Accessed: Sep. 18, 2022. [Online]. Available: https://conditionreportsonline.com.au/

[128] *Beal Trabajo*. 2020. Accessed: May 16, 2020. [Online]. Available: http://www.mawd.info/beal/Beal-Cuerdas-y-Equipo-para-Trabajo-y-Rescate.asp

[129] B. F. Spencer, V. Hoskere, and Y. Narazaki, “Advances in Computer Vision-Based Civil Infrastructure Inspection and Monitoring,” *Engineering*, vol. 5, no. 2, pp. 199–222, Apr. 2019, doi: 10.1016/j.eng.2018.11.030.

[130] “Construction Site Monitoring using Unmanned Aerial Vehicle.” Accessed: Sep. 28, 2022. [Online]. Available: https://www.equinoxsdrones.com/blog/construction-site-monitoring-using-unmanned-aerial-vehicle

[131] *Reinforcement Learning Bolsters Automated Detection of Concrete Cracks*. Accessed: Nov. 09, 2022. [Online]. Available: https://www.cmu.edu/news/stories/archives/2022/april/crack-detection.html

[132] Z. Yiyang, "The design of glass crack detection system based on image pre-processing technology," in *2014 IEEE 7th Joint International Information Technology and Artificial Intelligence Conference*, Dec. 2014, pp. 39–42. doi: 10.1109/ITAIC.2014.7065001.

[133] X. Zhou and X. Liu, "The Detection and Recognition Algorithm of Safety Glass Fragment," in *2007 International Conference on Mechatronics and Automation*, Aug. 2007, pp. 963–967. doi: 10.1109/ICMA.2007.4303677.

[134] Y. Fujita and Y. Hamamoto, "A robust automatic crack detection method from noisy concrete surfaces," *Machine Vision and Applications*, vol. 22, no. 2, pp. 245–254, Mar. 2011, doi: 10.1007/s00138-009-0244-5.

[135] G. Li, X. Zhao, K. Du, F. Ru, and Y. Zhang, "Recognition and evaluation of bridge cracks with modified active contour model and greedy search-based support vector machine," *Automation in Construction*, vol. 78, pp. 51–61, Jun. 2017, doi: 10.1016/j.autcon.2017.01.019.

[136] Ç. F. Özgenel and A. G. Sorguç, "Performance Comparison of Pretrained Convolutional Neural Networks on Crack Detection in Buildings," Jul. 2018. doi: 10.22260/ISARC2018/0094.

[137] Ç. F. Özgenel, "Concrete Crack Images for Classification." Mendeley, Jul. 23, 2019. doi: 10.17632/5Y9WDSG2ZT.2.

[138] S. Li and X. Zhao, "Image-Based Concrete Crack Detection Using Convolutional Neural Network and Exhaustive Search Technique," *Advances in Civil Engineering*, vol. 2019, pp. 1–12, Apr. 2019, doi: 10.1155/2019/6520620.

[139] C. Su and W. Wang, "Concrete Cracks Detection Using Convolutional NeuralNetwork Based on Transfer Learning," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–10, Oct. 2020, doi: 10.1155/2020/7240129.

[140] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *2016 IEEE International Conference on Image Processing (ICIP)*, Sep. 2016, pp. 3708–3712. doi: 10.1109/ICIP.2016.7533052.

[141] S. Bhardwaj and A. Mittal, "A Survey on Various Edge Detector Techniques," *Procedia Technology*, vol. 4, pp. 220–226, 2012, doi: 10.1016/j.protcy.2012.05.033.

[142] M. Stojmenović, A. Nayak, and J. Zunic, “Measuring linearity of planar point sets,” *Pattern Recognition*, vol. 41, no. 8, pp. 2503–2511, Aug. 2008, doi: 10.1016/j.patcog.2008.01.013.

[143] R. W. Emerson, “Causation and Pearson’s Correlation Coefficient,” *Journal of Visual Impairment & Blindness*, vol. 109, no. 3, pp. 242–244, May 2015, doi: 10.1177/0145482X1510900311.

[144] M. Worring and A. W. M. Smeulders, “Digital Curvature Estimation,” *CVGIP: Image Understanding*, vol. 58, no. 3, pp. 366–382, Nov. 1993, doi: 10.1006/ciun.1993.1048.

[145] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *arXiv:1409.1556 [cs]*, Apr. 2015, Accessed: Oct. 11, 2021. [Online]. Available: http://arxiv.org/abs/1409.1556

[146] H. Kaiming, Z. Xiangyu, R. Shaoqing, and S. Jian, “Deep Residual Learning for Image Recognition,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[147] A. I. Baba and C. Cornel, “‘Tumor cell morphology.’ Comparative oncology. , 2007.,” The Publishing House of the Romanian Academy, 2007.

[148] J. K. Hoang, W. K. Lee, M. Lee, D. Johnson, and S. Farrell, “US Features of Thyroid Malignancy: Pearls and Pitfalls,” *RadioGraphics*, vol. 27, no. 3, pp. 847–860, May 2007, doi: 10.1148/rg.273065038.

[149] F. N. Tessler *et al.*, “ACR Thyroid Imaging, Reporting and Data System (TI-RADS): White Paper of the ACR TI-RADS Committee,” *Journal of the American College of Radiology*, vol. 14, no. 5, pp. 587–595, May 2017, doi: 10.1016/j.jacr.2017.01.046.

[150] E. Lazarus, M. B. Mainiero, B. Schepps, S. L. Koelliker, and L. S. Livingston, “BI-RADS Lexicon for US and Mammography: Interobserver Variability and Positive Predictive Value,” *Radiology*, vol. 239, no. 2, pp. 385–391, May 2006, doi: 10.1148/radiol.2392042127.

[151] F. Nachbar *et al.*, “The ABCD rule of dermatoscopy,” *Journal of the American Academy of Dermatology*, vol. 30, no. 4, pp. 551–559, Apr. 1994, doi: 10.1016/S0190-9622(94)70061-3.

[152] E. G. Grant *et al.*, “Thyroid Ultrasound Reporting Lexicon: White Paper of the ACR Thyroid Imaging, Reporting and Data System (TIRADS) Committee,” *Journal of the American College of Radiology*, vol. 12, no. 12, pp. 1272–1279, Dec. 2015, doi: 10.1016/j.jacr.2015.07.011.

[153] A. Boezaart and B. Ihnatsenka, “Ultrasound: Basic understanding and learning the language,” *Int J Shoulder Surg*, vol. 4, no. 3, p. 55, 2010, doi: 10.4103/0973-6042.76960.

[154] H. A. Nugroho, E. L. Frannita, A. Nugroho, Zulfanahri, I. Ardiyanto, and L. Choridah, "Classification of thyroid nodules based on analysis of margin characteristic," in *2017 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, Jakarta, Oct. 2017, pp. 47–51. doi: 10.1109/IC3INA.2017.8251738.

[155] L. Pedraza, C. Vargas, F. Narváez, O. Durán, E. Muñoz, and E. Romero, "An open access thyroid ultrasound image database," Cartagena de Indias, Colombia, Jan. 2015, p. 92870W. doi: 10.1117/12.2073532.

[156] Y. Yan, W. Zhu, Y. Wu, and D. Zhang, "Fractal Dimension Differentiation between Benign and Malignant Thyroid Nodules from Ultrasonography," *Applied Sciences*, vol. 9, no. 7, p. 1494, Apr. 2019, doi: 10.3390/app9071494.

[157] J.-H. Lee *et al.*, "Computer-aided lesion diagnosis in B-mode ultrasound by border irregularity and multiple sonographic features," Lake Buena Vista (Orlando Area), Florida, USA, Feb. 2013, p. 86701O. doi: 10.1117/12.2007452.

[158] G. Zhang, S. Shin, W. Wang, C. Hruska, and H. D. Choi, "A new Fourier-based approach to measure irregularity of breast masses in mammograms," in *Proceedings of the 2012 ACM Research in Applied Computation Symposium on - RACS '12*, San Antonio, Texas, 2012, p. 153. doi: 10.1145/2401603.2401638.

[159] W. C. A. Pereira, A. V. Alvarenga, A. F. C. Infantosi, L. Macrini, and C. E. Pedreira, "A non-linear morphometric feature selection approach for breast tumor contour from ultrasonic images," *Computers in Biology and Medicine*, vol. 40, no. 11–12, pp. 912–918, Nov. 2010, doi: 10.1016/j.compbiomed.2010.10.003.

[160] Y.-H. Chou, C.-M. Tiu, G.-S. Hung, S.-C. Wu, T. Y. Chang, and H. K. Chiang, "Stepwise logistic regression analysis of tumor contour features for breast ultrasound diagnosis," *Ultrasound in Medicine & Biology*, vol. 27, no. 11, pp. 1493–1498, Nov. 2001, doi: 10.1016/S0301-5629(01)00466-5.

[161] F. Ouyang *et al.*, "Comparison between linear and nonlinear machine-learning algorithms for the classification of thyroid nodules," *European Journal of Radiology*, vol. 113, pp. 251–257, Apr. 2019, doi: 10.1016/j.ejrad.2019.02.029.

[162] J. Jaworek-Korjakowska, "Novel Method for Border Irregularity Assessment in Dermoscopic Color Images," *Computational and Mathematical Methods in Medicine*, vol. 2015, pp. 1–11, 2015, doi: 10.1155/2015/496202.

[163] V. Ng and A. Coldman, “Diagnosis of melanoma with fractal dimensions,” in *Proceedings of TENCON ’93. IEEE Region 10 International Conference on Computers, Communications and Automation*, Beijing, China, 1993, pp. 514–517. doi: 10.1109/TENCON.1993.320544.

[164] T. K. Lee, “MEASURING BORDER IRREGULARITY AND SHAPE OF CUTANEOUS MELANOCYTIC LESIONS,” SIMON FRASER UNIVERSITY, 2001. [Online]. Available: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.440.1861&rep=rep1&type=pdf

[165] C. S. Park *et al.*, “Observer variability in the sonographic evaluation of thyroid nodules,” *J. Clin. Ultrasound*, p. NA-NA, 2010, doi: 10.1002/jcu.20689.

[166] W. Gander, G. H. Golub, and R. Strebel, “Least-squares fitting of circles and ellipses,” *BIT*, vol. 34, no. 4, pp. 558–578, Dec. 1994, doi: 10.1007/BF01934268.

[167] T.-C. Wu, S. A. Belteton, J. Pack, D. B. Szymanski, and D. M. Umulis, “LobeFinder: A Convex Hull-Based Method for Quantitative Boundary Analyses of Lobed Plant Cells,” *Plant Physiol.*, vol. 171, no. 4, Art. no. 4, Aug. 2016, doi: 10.1104/pp.15.00972.

[168] S. Liu-Yu and M. Thonnat, “Description of object shapes by apparent boundary and convex hull,” *Pattern Recognition*, vol. 26, no. 1, Art. no. 1, Jan. 1993, doi: 10.1016/0031-3203(93)90091-A.

[169] K. G. Kim, S. W. Cho, S. J. Min, J. H. Kim, B. G. Min, and K. T. Bae, “Computerized scheme for assessing ultrasonographic features of breast masses1,” *Academic Radiology*, vol. 12, no. 1, Art. no. 1, Jan. 2005, doi: 10.1016/j.acra.2004.11.010.

[170] M. H. Jafari, S. Samavi, N. Karimi, S. M. R. Soroushmehr, K. Ward, and K. Najarian, “Automatic detection of melanoma using broad extraction of features from digital images,” in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Orlando, FL, USA, Aug. 2016, pp. 1357–1360. doi: 10.1109/EMBC.2016.7590959.

[171] O. Grove *et al.*, “Quantitative Computed Tomographic Descriptors Associate Tumor Shape Complexity and Intratumor Heterogeneity with Prognosis in Lung Adenocarcinoma,” *PLoS ONE*, vol. 10, no. 3, p. e0118261, Mar. 2015, doi: 10.1371/journal.pone.0118261.

[172] A. Asaad, “Persistent Homology for Image Analysis.,” the University of Buckingham, 2020. [Online]. Available: https://www.researchgate.net/publication/341136357_Persistent_Homology_for_Image_Analysis

[173] M. B. Villarino, “Ramanujan’s Perimeter of an Ellipse,” *arXiv:math/0506384*, Jun. 2005, Accessed: Mar. 08, 2022. [Online]. Available: http://arxiv.org/abs/math/0506384

[174] F. Mohammad, A. Alzoubi, H. Du, and S. Jassim, “Machine leaning assessment of border irregularity of thyroid nodules from ultrasound images,” in *Multimodal Image Exploitation and Learning 2022*, Orlando, United States, May 2022, p. 6. doi: 10.1117/12.2618470.

[175] K. Munir, H. Elahi, A. Ayub, F. Frezza, and A. Rizzi, “Cancer Diagnosis Using Deep Learning: A Bibliographic Review,” *Cancers*, vol. 11, no. 9, p. 1235, Aug. 2019, doi: 10.3390/cancers11091235.

[176] J. R. Burt *et al.*, “Deep learning beyond cats and dogs: recent advances in diagnosing breast cancer with deep neural networks,” *BJR*, p. 20170545, Apr. 2018, doi: 10.1259/bjr.20170545.

[177] H. D. Cheng, J. Shan, W. Ju, Y. Guo, and L. Zhang, “Automated breast cancer detection and classification using ultrasound images: A survey,” *Pattern Recognition*, vol. 43, no. 1, Art. no. 1, Jan. 2010, doi: 10.1016/j.patcog.2009.05.012.

[178] T. G. Dietterich, “Ensemble Methods in Machine Learning,” in *Multiple Classifier Systems*, vol. 1857, Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 1–15. doi: 10.1007/3-540-45014-9_1.

[179] Y. Liu, L. Ren, X. Cao, and Y. Tong, “Breast tumors recognition based on edge feature extraction using support vector machine,” *Biomedical Signal Processing and Control*, vol. 58, p. 101825, Apr. 2020, doi: 10.1016/j.bspc.2019.101825.

[180] M. K. Abd Ghani *et al.*, “Decision-level fusion scheme for nasopharyngeal carcinoma identification using machine learning techniques,” *Neural Comput & Applic*, vol. 32, no. 3, pp. 625–638, Feb. 2020, doi: 10.1007/s00521-018-3882-6.

[181] H. Du, *Data mining techniques and applications: an introduction*. Cengage Learning, 2010.

[182] E. Claridge, P. N. Hall, M. Keefe, and J. P. Allen, “Shape analysis for classification of malignant melanoma,” *Journal of Biomedical Engineering*, vol. 14, no. 3, Art. no. 3, May 1992, doi: 10.1016/0141-5425(92)90057-R.

# Appendix A

## A.1 Internal Dataset Statistics

The final dataset statistics diagram shows the relation between the number of ROI points and the size of the lesions (see scatter plot in Figure 5.7). The scatter plot shows strong variations in the number of ROI points between the lesions. It can be further observed that, in general, the higher the lesion's size, the higher its number of ROI points. There are also some outliers, such as the green dot on the right side having 136 ROI points, although its size is smaller than a handful of other lesions. Also, the purple dot on the top has the biggest size, above 180000 pixels, yet has a relatively small number of ROI points (around 30 points).

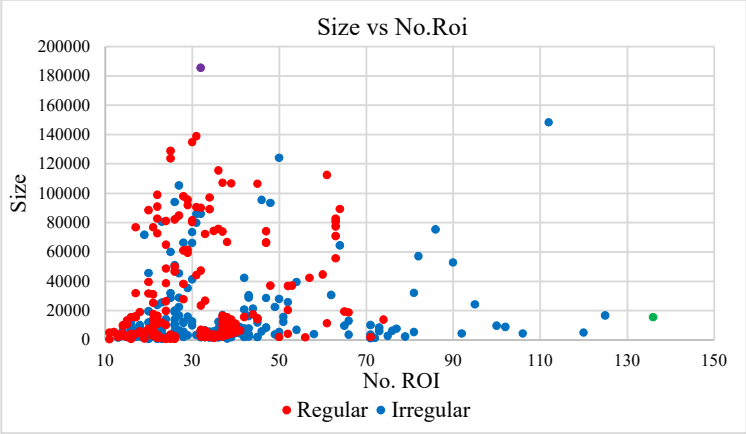
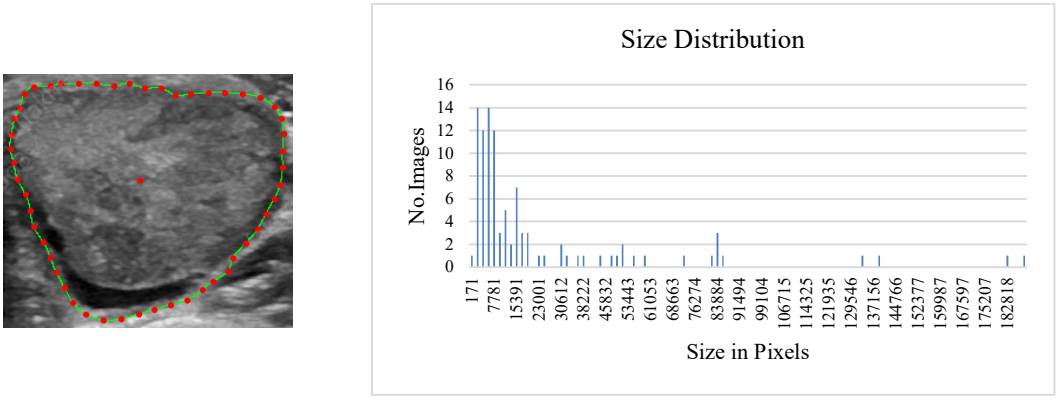



Figure 1: Distribution of the lesion sizes in relation to their number of ROI points.

## A.2 External Dataset Statistics

In the following, we show some external dataset statistics regarding the nodule size, the number of ROI points, and the nodule size distribution among the two classes. We can see from the histogram in Figure A.1 b that the majority of the thyroid nodules are of small sizes, around 171-7781 pixels. There are only two large nodules of around 182000 pixels, and the lesion size distribution is very similar to the internal dataset DS(395).




(a) (b)

Figure 2: (a) Thyroid cancer lesion (b) size distribution of the lesions across the external dataset.

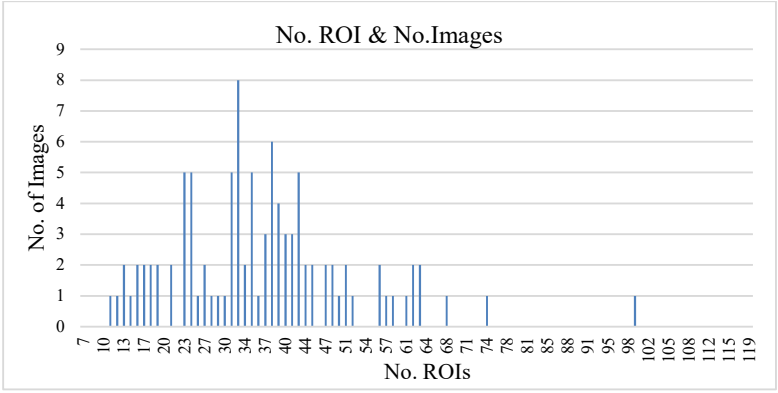



Figure 3: Distribution of the number of ROI points across the dataset DS(100).

Figure A.2 above demonstrates that the majority of the nodules from the external dataset have between 23 and 42 ROI points, whereas those from the internal dataset DS(395) were between 15 and 39 points, indicating that the number of ROI points in the external dataset is typically higher. Next, we present in the three charts in Figure A.3 the lesion size distribution among the two image classes for each of the three doctor's ground truths. Like in the internal dataset, there is no significant correlation between lesion sizes and the regular and irregular classes, although regular classes tend to have slightly larger sizes.

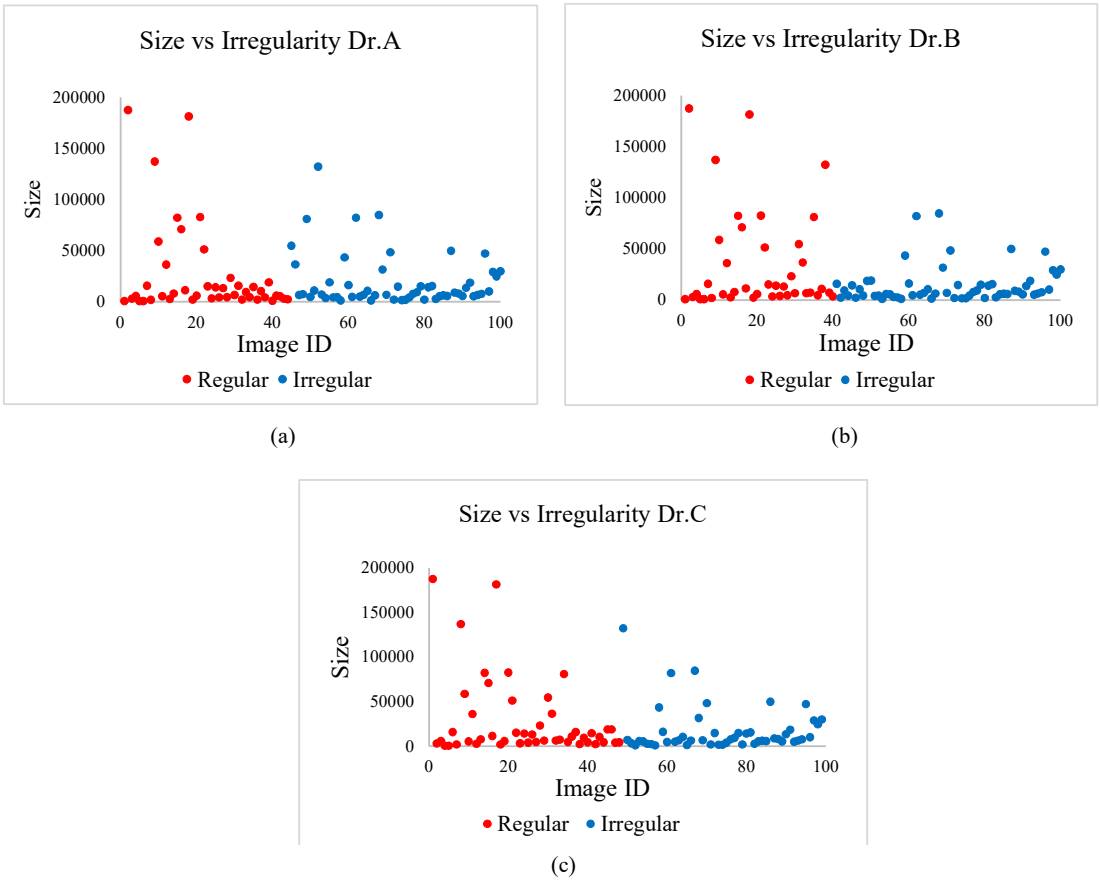



Figure 4: Lesion size vs irregularity class distribution across the external DS(100) dataset for three doctor's labels.

Finally, the three charts in Figure A.4 show the correlation between lesion size and the number of ROI points. The same observation as in the internal dataset can be seen, where the bigger the lesion size, the higher the number of ROI points. Additionally, we can see that the irregular cases typically have more ROI points since the border has more zigzags, which causes the radiologists to appoint more ROI points. However, one image (the dot on the right side) has the maximum number of ROI points of around 100 that is labelled as irregular by doctors A & C but labelled as regular by doctor B, which demonstrates the inter and intraobserver subjectivity variation of lesion border irregularity described in section 5.2.2.3.

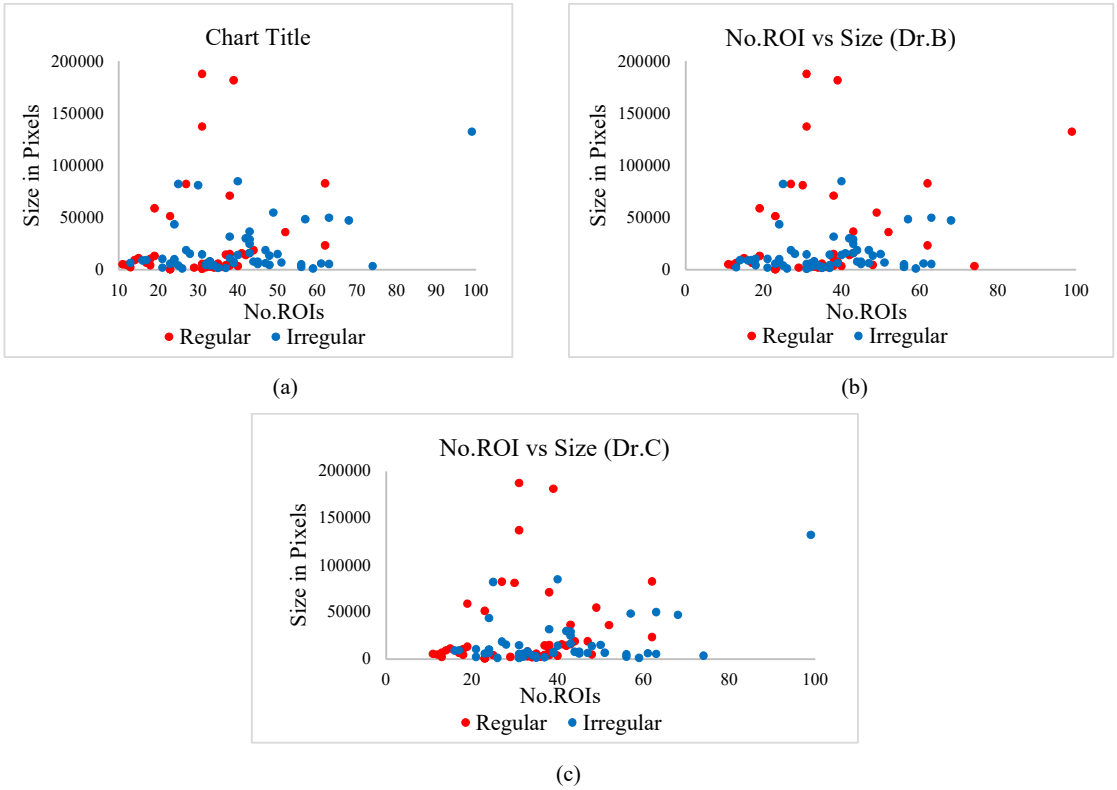



Figure 5: Lesion size vs the number of ROI points distributed across the external DS(100) for three doctors' ground truth.

# Appendix B

## B.1 ULBP based on the Whole Ribbon

Table 1: Whole lesion ribbon (radial distances ribbon construction)

| **ULBP(58) whole ribbon** (gamma='scale', kernel='poly', degree=9) | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | |
| Margin | Acc | Reg. | Irreg. | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 53 | 50 | 56 | 54 | 52 | 55 | 52 | 50 | 53 | 57 | 55 | 59 |
| 2 | 56 | 59 | 53 | 59 | 59 | 59 | 59 | 60 | 58 | 58 | 57 | 59 |
| 3 | 56 | 59 | 53 | 57 | 59 | 55 | 61 | 65 | 58 | 60 | 61 | 59 |
| **4** | **60** | **50** | **69** | **64** | **57** | **70** | **64** | **57** | **68** | **65** | **57** | **73** |
| 5 | 65 | 63 | 67 | 62 | 64 | 61 | 58 | 60 | 57 | 59 | 59 | 59 |
| 6 | 59 | 46 | 70 | 63 | 55 | 70 | 65 | 57 | 70 | 64 | 55 | 73 |
| 7 | 58 | 44 | 70 | 59 | 45 | 70 | 67 | 55 | 75 | 60 | 47 | 73 |
| 8 | 58 | 54 | 62 | 50 | 39 | 59 | 56 | 45 | 63 | 53 | 43 | 63 |
| 9 | 64 | 65 | 62 | 49 | 43 | 54 | 55 | 50 | 58 | 52 | 47 | 57 |
| 10 | 57 | 52 | 61 | 57 | 45 | 66 | 59 | 48 | 67 | 58 | 47 | 69 |
| 11 | 53 | 41 | 62 | 57 | 43 | 68 | 61 | 48 | 70 | 52 | 39 | 65 |
| 12 | 54 | 48 | 59 | 51 | 36 | 62 | 57 | 43 | 67 | 48 | 35 | 61 |
| 13 | 59 | 65 | 55 | 57 | 52 | 61 | 61 | 57 | 63 | 54 | 49 | 59 |
| 14 | 57 | 63 | 52 | 56 | 57 | 55 | 60 | 62 | 58 | 53 | 53 | 53 |
| 15 | 63 | 57 | 67 | 57 | 41 | 70 | 61 | 45 | 72 | 56 | 41 | 71 |
| 16 | 62 | 63 | 61 | 52 | 45 | 57 | 54 | 48 | 58 | 51 | 45 | 57 |
| 17 | 54 | 59 | 50 | 54 | 55 | 54 | 56 | 57 | 55 | 49 | 49 | 49 |
| **18** | **65** | **65** | **66** | **56** | **43** | **66** | **56** | **43** | **65** | **51** | **39** | **63** |
| 19 | 64 | 61 | 67 | 53 | 43 | 61 | 61 | 52 | 67 | 52 | 43 | 61 |
| 20 | 64 | 63 | 66 | 52 | 52 | 52 | 56 | 57 | 55 | 51 | 51 | 51 |
| 21 | 61 | 61 | 61 | 52 | 45 | 57 | 58 | 52 | 62 | 51 | 45 | 57 |
| 22 | 63 | 65 | 61 | 53 | 43 | 61 | 57 | 48 | 63 | 50 | 41 | 59 |
| 23 | 64 | 72 | 58 | 53 | 52 | 54 | 53 | 52 | 53 | 48 | 47 | 49 |
| 24 | 58 | 56 | 59 | 52 | 41 | 61 | 60 | 50 | 67 | 53 | 43 | 63 |

Table 2: Whole lesion ribbon (morphological ribbon construction).

| **ULBP(58) whole ribbon** (gamma='scale', kernel='poly', degree=9) | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | |
| Margin | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 49 | 33 | 62 | 59 | 39 | 75 | 55 | 32 | 70 | 56 | 37 | 75 |
| 2 | 58 | 50 | 66 | 57 | 55 | 59 | 59 | 57 | 60 | 54 | 51 | 57 |
| 3 | 56 | 37 | 72 | 62 | 43 | 77 | 62 | 43 | 75 | 57 | 39 | 75 |
| 4 | 56 | 41 | 69 | 54 | 41 | 64 | 60 | 48 | 68 | 57 | 45 | 69 |
| **5** | **59** | **54** | **64** | **57** | **50** | **62** | **61** | **55** | **65** | **60** | **53** | **67** |
| 6 | 59 | 50 | 67 | 56 | 43 | 66 | 58 | 45 | 67 | 55 | 43 | 67 |
| 7 | 58 | 54 | 62 | 57 | 48 | 64 | 61 | 52 | 67 | 58 | 49 | 67 |
| 8 | 60 | 54 | 66 | 55 | 45 | 62 | 55 | 45 | 62 | 54 | 45 | 63 |
| 9 | 58 | 44 | 69 | 59 | 36 | 77 | 65 | 43 | 80 | 60 | 39 | 80 |
| **10** | **65** | **63** | **67** | **57** | **52** | **61** | **61** | **57** | **63** | **54** | **49** | **59** |
| 11 | 53 | 48 | 56 | 58 | 48 | 66 | 64 | 55 | 70 | 55 | 45 | 65 |
| 12 | 61 | 57 | 64 | 50 | 36 | 61 | 54 | 40 | 63 | 49 | 37 | 61 |
| 13 | 58 | 54 | 62 | 57 | 39 | 71 | 59 | 40 | 72 | 52 | 35 | 69 |
| 14 | 64 | 63 | 66 | 56 | 50 | 61 | 58 | 52 | 62 | 53 | 47 | 59 |
| 15 | 59 | 59 | 59 | 55 | 50 | 59 | 59 | 55 | 62 | 54 | 49 | 59 |
| 16 | 61 | 63 | 59 | 59 | 50 | 66 | 63 | 55 | 68 | 56 | 47 | 65 |
| 17 | 55 | 52 | 58 | 55 | 48 | 61 | 59 | 52 | 63 | 52 | 45 | 59 |
| 18 | 58 | 56 | 59 | 55 | 45 | 62 | 59 | 50 | 65 | 52 | 43 | 61 |
| 19 | 63 | 63 | 62 | 54 | 45 | 61 | 60 | 52 | 65 | 53 | 45 | 61 |
| 20 | 61 | 59 | 62 | 52 | 43 | 59 | 60 | 52 | 65 | 51 | 43 | 59 |
| 21 | 58 | 59 | 58 | 55 | 45 | 62 | 63 | 55 | 68 | 54 | 45 | 63 |
| 22 | 59 | 65 | 55 | 56 | 55 | 57 | 56 | 55 | 57 | 51 | 49 | 53 |
| 23 | 58 | 67 | 52 | 51 | 48 | 54 | 51 | 48 | 53 | 46 | 43 | 49 |
| 24 | 56 | 44 | 66 | 51 | 34 | 64 | 55 | 38 | 67 | 48 | 33 | 63 |

## B.2 Ribbon Sectors-based ULBP

Table 3: Ribbon sectors based ULBP using morphological ribbon construction.

| **ULBP(58) ribbon sectors** SVM (C=100, degree=1, gamma=0.1, kernel='rbf') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Margin | Acc | Reg | Irreg | Acc | Acc | Reg | Irreg | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 58 | 59 | 58 | 53 | 45 | 59 | 57 | 50 | 62 | 56 | 49 | 63 | 59 | 50 | 65 |
| 2 | 55 | 59 | 52 | 61 | 59 | 62 | 63 | 62 | 63 | 68 | 65 | 71 | 68 | 63 | 72 |
| 3 | 63 | 57 | 67 | 72 | 61 | 80 | 68 | 57 | 75 | 69 | 57 | 80 | 77 | 60 | 88 |
| 4 | 59 | 54 | 64 | 63 | 50 | 73 | 61 | 48 | 70 | 62 | 49 | 75 | 67 | 50 | 79 |
| 5 | 59 | 48 | 69 | 70 | 55 | 82 | 70 | 55 | 80 | 71 | 55 | 86 | 78 | 60 | 91 |
| 6 | 61 | 54 | 67 | 67 | 52 | 79 | 69 | 55 | 78 | 70 | 55 | 84 | 75 | 60 | 86 |
| 7 | 64 | 61 | 66 | 67 | 55 | 77 | 69 | 57 | 77 | 70 | 57 | 82 | 75 | 63 | 84 |
| 8 | 64 | 63 | 64 | 69 | 57 | 79 | 71 | 60 | 78 | 72 | 59 | 84 | 78 | 63 | 88 |
| 9 | 62 | 54 | 69 | 64 | 52 | 73 | 74 | 65 | 80 | 71 | 59 | 82 | 77 | 63 | 86 |
| 10 | 63 | 61 | 64 | 66 | 59 | 71 | 68 | 62 | 72 | 67 | 59 | 75 | 74 | 67 | 79 |
| 11 | 61 | 56 | 66 | 64 | 55 | 71 | 66 | 57 | 72 | 65 | 55 | 75 | 71 | 60 | 79 |
| 12 | 64 | 63 | 66 | 68 | 57 | 77 | 68 | 57 | 75 | 67 | 55 | 78 | 75 | 63 | 84 |
| 13 | 69 | 70 | 67 | 66 | 57 | 73 | 70 | 62 | 75 | 69 | 59 | 78 | 75 | 67 | 81 |
| 14 | 69 | 70 | 69 | 66 | 55 | 75 | 68 | 57 | 75 | 67 | 55 | 78 | 74 | 63 | 81 |
| 15 | 66 | 65 | 67 | 68 | 57 | 77 | 72 | 62 | 78 | 71 | 59 | 82 | 78 | 67 | 86 |
| 16 | 70 | 70 | 70 | 68 | 61 | 73 | 66 | 60 | 70 | 69 | 61 | 76 | 74 | 67 | 79 |
| 17 | 64 | 65 | 64 | 67 | 59 | 73 | 71 | 65 | 75 | 70 | 61 | 78 | 77 | 70 | 81 |
| 18 | 68 | 70 | 66 | 65 | 57 | 71 | 67 | 60 | 72 | 66 | 57 | 75 | 73 | 63 | 79 |
| 19 | 69 | 67 | 72 | 63 | 52 | 71 | 61 | 50 | 68 | 62 | 51 | 73 | 67 | 57 | 74 |
| **20** | **72** | **63** | **80** | **67** | **55** | **77** | **73** | **62** | **80** | **70** | **57** | **82** | **78** | **67** | **86** |
| 21 | 69 | 63 | 75 | 68 | 57 | 77 | 70 | 60 | 77 | 69 | 57 | 80 | 77 | 67 | 84 |
| **22** | **72** | **70** | **73** | **67** | **55** | **77** | **69** | **57** | **77** | **68** | **55** | **80** | **75** | **67** | **81** |
| 23 | 68 | 69 | 67 | 69 | 57 | 79 | 67 | 55 | 75 | 68 | 55 | 80 | 75 | 63 | 84 |
| 24 | 69 | 69 | 69 | 67 | 55 | 77 | 65 | 52 | 73 | 66 | 53 | 78 | 73 | 60 | 81 |

## B.3 ULBP based on Code Groups

Table 4: G3 tested on the internal and external testing set.

| **ULBP(G3) ribbon sectors** SVM (C=100, degree=1, gamma=0.1, kernel='rbf') | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | |
| Margin | ACC | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 55 | 54 | 56 | 58 | 61 | 55 | 54 | 57 | 52 | 55 | 57 | 53 |
| 2 | 64 | 70 | 59 | 65 | 64 | 66 | 61 | 60 | 62 | 66 | 63 | 69 |
| 3 | 61 | 57 | 64 | 59 | 50 | 66 | 63 | 55 | 68 | 64 | 55 | 73 |
| 4 | 64 | 57 | 69 | 69 | 64 | 73 | 55 | 48 | 60 | 62 | 55 | 69 |
| 5 | 64 | 69 | 59 | 65 | 64 | 66 | 65 | 65 | 65 | 66 | 63 | 69 |
| 6 | 60 | 59 | 61 | 66 | 55 | 75 | 62 | 50 | 70 | 69 | 57 | 80 |
| 7 | 68 | 74 | 62 | 63 | 57 | 68 | 65 | 60 | 68 | 68 | 61 | 75 |
| 8 | 69 | 74 | 64 | 65 | 59 | 70 | 71 | 68 | 73 | 72 | 65 | 78 |
| 9 | 65 | 67 | 64 | 62 | 57 | 66 | 60 | 55 | 63 | 63 | 57 | 69 |
| 10 | 69 | 69 | 69 | 61 | 57 | 64 | 57 | 52 | 60 | 60 | 55 | 65 |
| **11** | **72** | **72** | **72** | **64** | **52** | **73** | **60** | **48** | **68** | **61** | **49** | **73** |
| 12 | 68 | 69 | 67 | 66 | 55 | 75 | 62 | 50 | 70 | 63 | 51 | 75 |
| 13 | 69 | 74 | 64 | 66 | 55 | 75 | 60 | 48 | 68 | 63 | 51 | 75 |
| 14 | 69 | 63 | 73 | 60 | 43 | 73 | 62 | 45 | 73 | 63 | 47 | 78 |
| 15 | 70 | 72 | 69 | 64 | 61 | 66 | 60 | 57 | 62 | 67 | 63 | 71 |
| 16 | 67 | 76 | 59 | 59 | 64 | 55 | 59 | 65 | 55 | 62 | 65 | 59 |
| 17 | 69 | 76 | 64 | 66 | 59 | 71 | 64 | 57 | 68 | 69 | 61 | 76 |
| 18 | 63 | 67 | 59 | 64 | 55 | 71 | 66 | 57 | 72 | 67 | 57 | 76 |
| 19 | 67 | 72 | 62 | 64 | 57 | 70 | 66 | 60 | 70 | 67 | 59 | 75 |
| 20 | 63 | 67 | 59 | 65 | 57 | 71 | 65 | 57 | 70 | 68 | 59 | 76 |
| 21 | 60 | 65 | 56 | 59 | 52 | 64 | 63 | 57 | 67 | 64 | 57 | 71 |
| 22 | 64 | 63 | 64 | 62 | 52 | 70 | 62 | 52 | 68 | 65 | 55 | 75 |
| 23 | 67 | 72 | 62 | 66 | 57 | 73 | 68 | 60 | 73 | 69 | 59 | 78 |
| 24 | 66 | 69 | 64 | 65 | 52 | 75 | 65 | 52 | 73 | 66 | 53 | 78 |

## B.4 Model Training using Multiple Generic Margin Widths

### Testing Model 1 on the external dataset:

Table 5: Model1 testing on the external dataset.

| External Testing Model 1 | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| Margin | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 50 | 57 | 45 | 60 | 70 | 53 | 55 | 61 | 49 | 58 | 70 | 49 |
| 2 | 57 | 48 | 64 | 65 | 57 | 70 | 62 | 53 | 71 | 66 | 57 | 72 |
| 3 | 62 | 50 | 71 | 66 | 55 | 73 | 65 | 53 | 76 | 70 | 53 | 81 |
| 4 | 65 | 52 | 75 | 67 | 55 | 75 | 68 | 55 | 80 | 73 | 57 | 84 |
| 5 | 68 | 57 | 77 | 66 | 55 | 73 | 69 | 57 | 80 | 74 | 60 | 84 |
| 6 | 68 | 57 | 77 | 70 | 60 | 77 | 73 | 61 | 84 | 77 | 63 | 86 |
| 7 | 69 | 57 | 79 | 71 | 60 | 78 | 72 | 59 | 84 | 77 | 63 | 86 |
| 8 | 69 | 64 | 73 | 71 | 68 | 73 | 72 | 65 | 78 | 77 | 70 | 81 |
| 9 | 69 | 61 | 75 | 73 | 68 | 77 | 72 | 63 | 80 | 78 | 70 | 84 |
| 10 | 68 | 61 | 73 | 74 | 70 | 77 | 73 | 65 | 80 | 78 | 73 | 81 |
| 11 | 68 | 61 | 73 | 72 | 68 | 75 | 71 | 63 | 78 | 77 | 70 | 81 |
| 12 | 69 | 59 | 77 | 75 | 68 | 80 | 72 | 61 | 82 | 79 | 70 | 86 |
| 13 | 71 | 64 | 77 | 73 | 68 | 77 | 74 | 65 | 82 | 79 | 70 | 86 |
| 14 | 69 | 64 | 73 | 71 | 68 | 73 | 72 | 65 | 78 | 77 | 70 | 81 |
| 15 | 68 | 64 | 71 | 74 | 72 | 75 | 73 | 67 | 78 | 78 | 73 | 81 |
| 16 | 74 | 68 | 79 | 74 | 70 | 77 | 77 | 69 | 84 | 82 | 73 | 88 |
| **17** | **74** | **70** | **77** | **74** | **72** | **75** | **77** | **71** | **82** | **82** | **73** | **88** |
| 18 | 71 | 68 | 73 | 71 | 70 | 72 | 74 | 69 | 78 | 78 | 70 | 84 |
| 19 | 74 | 68 | 79 | 72 | 68 | 75 | 75 | 67 | 82 | 81 | 70 | 88 |
| 20 | 71 | 64 | 77 | 69 | 62 | 73 | 72 | 63 | 80 | 77 | 63 | 86 |
| 21 | 71 | 68 | 73 | 69 | 68 | 70 | 72 | 67 | 76 | 77 | 70 | 81 |
| 22 | 69 | 59 | 77 | 71 | 62 | 77 | 72 | 61 | 82 | 77 | 63 | 86 |
| 23 | 67 | 61 | 71 | 67 | 62 | 70 | 68 | 61 | 75 | 73 | 63 | 79 |
| 24 | 66 | 61 | 70 | 66 | 62 | 68 | 69 | 63 | 75 | 73 | 63 | 79 |

### Testing Model 2 on the External Dataset:

Table 6: Model2 testing on the external dataset.

| External Testing Model 2 | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Doctor A | | | Doctor B | | | Doctor C | | | Agreed Cases | | |
| Margin | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 57 | 39 | 71 | 63 | 45 | 75 | 60 | 43 | 76 | 63 | 43 | 77 |
| 2 | 59 | 39 | 75 | 61 | 40 | 75 | 58 | 39 | 76 | 63 | 40 | 79 |
| 3 | 59 | 32 | 80 | 59 | 30 | 78 | 58 | 33 | 82 | 62 | 33 | 81 |
| 4 | 62 | 34 | 84 | 60 | 30 | 80 | 61 | 35 | 86 | 64 | 37 | 84 |
| 5 | 61 | 39 | 79 | 59 | 35 | 75 | 60 | 39 | 80 | 63 | 40 | 79 |
| 6 | 63 | 39 | 82 | 63 | 38 | 80 | 64 | 41 | 86 | 67 | 43 | 84 |
| 7 | 62 | 39 | 80 | 66 | 43 | 82 | 65 | 43 | 86 | 68 | 47 | 84 |
| 8 | 65 | 45 | 80 | 67 | 48 | 80 | 66 | 47 | 84 | 71 | 53 | 84 |
| 9 | 61 | 36 | 80 | 65 | 40 | 82 | 64 | 41 | 86 | 67 | 43 | 84 |
| 10 | 64 | 41 | 82 | 66 | 43 | 82 | 65 | 43 | 86 | 70 | 47 | 86 |
| 11 | 62 | 41 | 79 | 68 | 48 | 82 | 67 | 47 | 86 | 70 | 47 | 86 |
| 12 | 63 | 45 | 77 | 67 | 50 | 78 | 68 | 51 | 84 | 70 | 50 | 84 |
| 13 | 63 | 48 | 75 | 63 | 48 | 73 | 66 | 51 | 80 | 67 | 50 | 79 |
| 14 | 62 | 50 | 71 | 64 | 52 | 72 | 67 | 55 | 78 | 67 | 53 | 77 |
| 15 | 63 | 52 | 71 | 67 | 57 | 73 | 68 | 57 | 78 | 70 | 60 | 77 |
| 16 | 66 | 59 | 71 | 68 | 62 | 72 | 69 | 61 | 76 | 73 | 67 | 77 |
| **17** | **69** | **64** | **73** | **69** | **65** | **72** | **72** | **65** | **78** | **75** | **70** | **79** |
| 18 | 68 | 66 | 70 | 66 | 65 | 67 | 69 | 65 | 73 | **73** | **70** | **74** |
| 19 | 67 | 66 | 68 | 65 | 65 | 65 | 68 | 65 | 71 | 71 | 70 | 72 |
| 20 | 67 | 68 | 66 | 63 | 65 | 62 | 68 | 67 | 69 | 70 | 70 | 70 |
| 21 | 66 | 66 | 66 | 64 | 65 | 63 | 69 | 67 | 71 | 70 | 70 | 70 |
| 22 | 62 | 64 | 61 | 58 | 60 | 57 | 63 | 63 | 63 | 63 | 63 | 63 |
| 23 | 61 | 64 | 59 | 57 | 60 | 55 | 62 | 63 | 61 | 62 | 63 | 60 |
| 24 | 61 | 64 | 59 | 55 | 57 | 53 | 60 | 61 | 59 | 60 | 60 | 60 |

## B.5 Model Training Depending on Different Lesion Size Categories

Table 7: ULBP method trained and tested on three subsets of the dataset (small, medium, and large size lesions).

| Small Size ULBP (58), SVM-poly | | | |
|---|---|---|---|
| Margin | Acc | Reg | Irreg |
| 1 | 62 | 19 | 92 |
| 2 | 65 | 40 | 83 |
| 3 | 65 | 58 | 69 |
| 4 | 70 | 68 | 71 |
| 5 | 72 | 66 | 76 |
| 6 | 71 | 70 | 72 |
| 7 | 71 | 68 | 72 |
| 8 | 70 | 64 | 73 |
| 9 | 71 | 68 | 73 |
| 10 | 75 | 73 | 76 |
| **11** | **77** | **75** | **79** |
| 12 | 74 | 73 | 73 |
| 13 | 74 | 73 | 75 |
| 14 | 73 | 71 | 75 |
| 15 | 70 | 69 | 71 |
| 16 | 70 | 69 | 69 |
| 17 | 71 | 68 | 72 |
| 18 | 69 | 64 | 72 |
| 19 | 71 | 68 | 72 |
| 20 | 68 | 60 | 73 |
| 21 | 67 | 58 | 73 |
| 22 | 70 | 65 | 73 |
| 23 | 69 | 64 | 72 |
| 24 | 71 | 66 | 73 |

| Medium Size ULBP (58), SVM-poly | | | |
|---|---|---|---|
| Margin | Acc | Reg | Irreg |
| 1 | 63 | 8 | 93 |
| 2 | 67 | 19 | 93 |
| 3 | 66 | 23 | 90 |
| 4 | 65 | 31 | 85 |
| 5 | 66 | 35 | 84 |
| 6 | 70 | 43 | 85 |
| 7 | 70 | 45 | 84 |
| 8 | 71 | 47 | 84 |
| 9 | 74 | 53 | 85 |
| 10 | 76 | 53 | 89 |
| 11 | 76 | 51 | 90 |
| 12 | 76 | 51 | 90 |
| 13 | 75 | 53 | 88 |
| **14** | **76** | **57** | **86** |
| 15 | 76 | 57 | 86 |
| 16 | 75 | 54 | 86 |
| 17 | 75 | 54 | 88 |
| 18 | 74 | 49 | 88 |
| 19 | 74 | 49 | 88 |
| 20 | 75 | 47 | 90 |
| 21 | 74 | 45 | 90 |
| 22 | 72 | 45 | 86 |
| 23 | 70 | 45 | 84 |
| 24 | 70 | 45 | 84 |

| Large Size ULBP (58), SVM-poly | | | |
|---|---|---|---|
| Margin | Acc | Reg | Irreg |
| 1 | 66 | 87 | 33 |
| 2 | 66 | 88 | 31 |
| 3 | 68 | 86 | 41 |
| 4 | 71 | 87 | 47 |
| 5 | 72 | 87 | 48 |
| 6 | 73 | 87 | 51 |
| 7 | 70 | 86 | 45 |
| 8 | 69 | 85 | 45 |
| 9 | 72 | 86 | 49 |
| 10 | 74 | 88 | 53 |
| 11 | 72 | 87 | 49 |
| 12 | 73 | 87 | 51 |
| **13** | **75** | **89** | **53** |
| 14 | 72 | 88 | 49 |
| 15 | 72 | 88 | 49 |
| 16 | 73 | 90 | 47 |
| 17 | 75 | 90 | 51 |
| 18 | 75 | 90 | 51 |
| 19 | 72 | 89 | 47 |
| 20 | 72 | 90 | 45 |
| 21 | 72 | 90 | 45 |
| 22 | 70 | 87 | 43 |
| 23 | 71 | 89 | 43 |
| 24 | 72 | 87 | 48 |

Table 8: ULBP method trained and tested on a subset of the dataset (small size lesions).

| **Small Size**<br>**ULBP (58), protocol1**<br>SVM (C=100, degree=1, gamma=0.1, kernel='rbf') | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | |
| Margin | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 64 | 19 | 96 | 53 | 24 | 88 | 63 | 25 | 86 | 53 | 21 | 84 |
| 2 | 65 | 40 | 82 | 52 | 31 | 78 | 60 | 33 | 77 | 51 | 28 | 74 |
| 3 | 71 | 60 | 79 | 50 | 34 | 69 | 58 | 37 | 70 | 54 | 36 | 71 |
| 4 | 69 | 55 | 79 | 50 | 35 | 68 | 57 | 39 | 69 | 55 | 39 | 71 |
| 5 | 69 | 58 | 78 | 51 | 36 | 68 | 55 | 36 | 66 | 53 | 38 | 69 |
| 6 | 68 | 65 | 71 | 49 | 36 | 66 | 55 | 37 | 66 | 54 | 39 | 68 |
| 7 | 70 | 66 | 73 | 48 | 35 | 63 | 54 | 36 | 65 | 52 | 37 | 66 |
| 8 | 72 | 66 | 75 | 49 | 38 | 62 | 55 | 40 | 63 | 52 | 40 | 64 |
| 9 | 74 | 68 | 79 | 50 | 39 | 63 | 54 | 40 | 63 | 53 | 41 | 65 |
| 10 | 76 | 71 | 80 | 51 | 40 | 63 | 54 | 40 | 62 | 53 | 41 | 64 |
| **11** | **76** | **74** | **78** | **52** | **42** | **64** | **54** | **41** | **62** | **54** | **43** | **65** |
| 12 | 76 | 73 | 78 | 52 | 40 | 66 | 56 | 41 | 65 | 56 | 44 | 69 |
| 13 | 75 | 71 | 78 | 53 | 42 | 66 | 56 | 43 | 64 | 56 | 44 | 67 |
| 14 | 72 | 69 | 74 | 51 | 42 | 63 | 56 | 44 | 63 | 56 | 46 | 67 |
| 15 | 72 | 66 | 75 | 55 | 44 | 68 | 58 | 45 | 66 | 60 | 48 | 71 |
| 16 | 73 | 71 | 74 | 52 | 40 | 67 | 55 | 40 | 65 | 57 | 44 | 70 |
| 17 | 72 | 68 | 74 | 52 | 37 | 69 | 56 | 37 | 67 | 60 | 45 | 76 |
| 18 | 69 | 63 | 73 | 52 | 39 | 68 | 57 | 40 | 66 | 60 | 46 | 74 |
| 19 | 71 | 69 | 72 | 52 | 41 | 66 | 54 | 40 | 63 | 57 | 45 | 69 |
| 20 | 70 | 66 | 73 | 53 | 44 | 64 | 55 | 43 | 62 | 58 | 48 | 68 |
| 21 | 68 | 61 | 73 | 53 | 45 | 64 | 54 | 43 | 61 | 59 | 49 | 68 |
| 22 | 70 | 63 | 74 | 53 | 44 | 64 | 54 | 41 | 61 | 58 | 48 | 68 |
| 23 | 70 | 64 | 74 | 54 | 44 | 66 | 52 | 39 | 60 | 57 | 46 | 67 |
| 24 | 70 | 64 | 74 | 54 | 45 | 66 | 53 | 40 | 60 | 57 | 47 | 67 |

Table 9: ULBP method trained and tested on a subset of the dataset (large-size lesions).

| **Large Size**<br>**ULBP (58), protocol1**<br>SVM (C=100, degree=1, gamma=0.1, kernel='rbf') | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | |
| Margin | Acc. | Reg. | Irreg. | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 61 | 71 | 51 | 58 | 58 | 58 | 59 | 58 | 59 | 59 | 57 | 61 |
| 2 | 64 | 73 | 56 | 65 | 61 | 68 | 64 | 58 | 69 | 68 | 61 | 74 |
| 3 | 69 | 73 | 65 | 67 | 65 | 68 | 69 | 66 | 71 | 73 | 68 | 78 |
| 4 | 66 | 68 | 64 | 67 | 67 | 68 | 66 | 63 | 68 | 70 | 66 | 74 |
| 5 | 68 | 70 | 67 | 71 | 71 | 71 | 71 | 69 | 73 | 74 | 69 | 78 |
| 6 | 69 | 70 | 69 | 70 | 65 | 73 | 69 | 62 | 74 | 73 | 64 | 81 |
| 7 | 71 | 73 | 69 | 69 | 64 | 72 | 66 | 58 | 71 | 71 | 62 | 79 |
| 8 | 68 | 72 | 65 | 68 | 65 | 69 | 68 | 64 | 71 | 71 | 66 | 77 |
| 9 | 71 | 76 | 66 | 68 | 63 | 71 | 68 | 62 | 73 | 70 | 61 | 77 |
| 10 | 69 | 70 | 70 | 67 | 61 | 70 | 69 | 62 | 74 | 70 | 62 | 78 |
| 11 | 70 | 71 | 70 | 68 | 61 | 72 | 69 | 62 | 75 | 70 | 61 | 79 |
| 12 | 70 | 70 | 70 | 69 | 63 | 73 | 70 | 62 | 76 | 72 | 62 | 81 |
| 13 | 69 | 69 | 69 | 72 | 70 | 73 | 72 | 68 | 75 | 75 | 68 | 81 |
| 14 | 69 | 69 | 70 | 73 | 70 | 74 | 73 | 69 | 77 | 76 | 68 | 83 |
| 15 | 68 | 69 | 68 | 73 | 70 | 75 | 73 | 68 | 77 | 75 | 68 | 83 |
| 16 | 70 | 70 | 71 | 74 | 73 | 75 | 72 | 67 | 75 | 77 | 70 | 83 |
| 17 | 70 | 70 | 71 | 71 | 66 | 74 | 72 | 65 | 77 | 75 | 66 | 83 |
| 18 | 71 | 72 | 71 | 72 | 67 | 74 | 71 | 65 | 76 | 75 | 67 | 83 |
| 19 | 70 | 71 | 70 | 71 | 67 | 74 | 71 | 65 | 75 | 75 | 67 | 83 |
| 20 | 72 | 72 | 73 | 70 | 65 | 73 | 71 | 64 | 75 | 74 | 65 | 81 |
| **21** | **73** | **73** | **74** | **71** | **67** | **73** | **70** | **65** | **74** | **74** | **66** | **81** |
| 22 | 71 | 69 | 74 | 71 | 70 | 72 | 71 | 67 | 73 | 75 | 69 | 80 |
| 23 | 72 | 69 | 74 | 71 | 67 | 73 | 72 | 66 | 76 | 75 | 67 | 82 |
| 24 | 71 | 70 | 72 | 72 | 68 | 74 | 73 | 68 | 77 | 75 | 68 | 83 |

## B.6 HOG-based Method

Table 10: experimental results of the HOG method based on ribbon using nine sectors on internal and external testing sets.

| **HOG ribbon sectors** SVM(C=1, degree=5, gamma=1, kernel='poly') | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Margin | Acc | Reg | Irreg | Acc | Acc | Reg | Irreg | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 58 | 69 | 50 | 41 | 45 | 38 | 49 | 55 | 45 | 46 | 51 | 41 | 44 | 47 | 42 |
| 2 | 61 | 61 | 61 | 50 | 45 | 54 | 52 | 48 | 55 | 53 | 49 | 57 | 52 | 47 | 56 |
| 3 | 57 | 61 | 53 | 55 | 48 | 61 | 55 | 48 | 60 | 54 | 47 | 61 | 56 | 50 | 60 |
| 4 | 57 | 44 | 67 | 59 | 50 | 66 | 57 | 48 | 63 | 54 | 45 | 63 | 59 | 57 | 60 |
| 5 | 53 | 44 | 61 | 61 | 45 | 73 | 63 | 48 | 73 | 60 | 45 | 75 | 64 | 50 | 74 |
| 6 | 58 | 46 | 67 | 62 | 43 | 77 | 66 | 48 | 78 | 63 | 45 | 80 | 68 | 50 | 81 |
| 7 | 56 | 56 | 56 | 62 | 43 | 77 | 66 | 48 | 78 | 63 | 45 | 80 | 68 | 50 | 81 |
| 8 | 64 | 61 | 66 | 58 | 43 | 70 | 56 | 40 | 67 | 57 | 43 | 71 | 59 | 40 | 72 |
| 9 | 59 | 52 | 66 | 63 | 39 | 82 | 73 | 50 | 88 | 68 | 45 | 90 | 74 | 50 | 91 |
| 10 | 59 | 52 | 66 | 65 | 48 | 79 | 69 | 52 | 80 | 66 | 49 | 82 | 73 | 53 | 86 |
| 11 | 56 | 39 | 70 | 61 | 34 | 82 | 69 | 43 | 87 | 66 | 41 | 90 | 70 | 40 | 91 |
| 12 | 60 | 44 | 73 | 63 | 34 | 86 | 71 | 43 | 90 | 66 | 39 | 92 | 73 | 43 | 93 |
| 13 | 60 | 43 | 75 | 60 | 30 | 84 | 70 | 40 | 90 | 63 | 35 | 90 | 70 | 40 | 91 |
| 14 | 60 | 56 | 64 | 59 | 43 | 71 | 63 | 48 | 73 | 62 | 47 | 76 | 66 | 50 | 77 |
| 15 | 61 | 50 | 70 | 60 | 30 | 84 | 72 | 43 | 92 | 63 | 35 | 90 | 71 | 43 | 91 |
| 16 | 64 | 54 | 72 | 65 | 45 | 80 | 69 | 50 | 82 | 66 | 47 | 84 | 74 | 53 | 88 |
| 17 | 60 | 56 | 64 | 59 | 45 | 70 | 65 | 52 | 73 | 62 | 49 | 75 | 67 | 50 | 79 |
| 18 | 63 | 48 | 75 | 61 | 32 | 84 | 75 | 48 | 93 | 66 | 39 | 92 | 74 | 47 | 93 |
| 19 | 63 | 59 | 66 | 60 | 45 | 71 | 66 | 52 | 75 | 63 | 49 | 76 | 68 | 50 | 81 |
| 20 | 62 | 56 | 67 | 59 | 43 | 71 | 65 | 50 | 75 | 62 | 47 | 76 | 67 | 47 | 81 |
| **21** | **68** | **57** | **77** | **57** | **36** | **73** | **59** | **38** | **73** | **58** | **39** | **76** | **62** | **37** | **79** |
| 22 | 64 | 61 | 66 | 59 | 43 | 71 | 65 | 50 | 75 | 62 | 47 | 76 | 67 | 47 | 81 |
| 23 | 66 | 61 | 70 | 55 | 36 | 70 | 65 | 48 | 77 | 62 | 45 | 78 | 64 | 43 | 79 |
| **24** | **66** | **61** | **70** | **57** | **39** | **71** | **67** | **50** | **78** | **64** | **47** | **80** | **67** | **47** | **81** |

## B.7 Histogram of Linearity

Table 11: Whole ribbon(morphological) using protocol3.

| **HOL Whole Ribbon** SVM(kernel='rbf') **Internal Testing** | | | |
|---|---|---|---|
| Margin | Acc. | Reg. | Irreg. |
| 10 | 53 | 65 | 44 |
| 11 | 52 | 26 | 73 |
| 12 | 53 | 6 | 92 |
| 13 | 53 | 54 | 52 |
| 14 | 48 | 54 | 44 |
| 15 | 48 | 30 | 64 |
| 16 | 47 | 57 | 38 |
| 17 | 53 | 61 | 47 |
| 18 | 48 | 59 | 39 |
| 19 | 56 | 9 | 95 |
| 20 | 54 | 6 | 95 |
| 21 | 57 | 59 | 55 |
| 22 | 57 | 11 | 95 |
| 23 | 49 | 63 | 38 |
| 24 | 43 | 52 | 36 |

| **HOL Ribbon Sectors** SVM( kernel='rbf') **Internal Testing** | | | |
|---|---|---|---|
| Margin | Acc. | Reg. | Irreg. |
| 10 | 58 | 26 | 84 |
| 11 | 57 | 23 | 86 |
| 12 | 55 | 42 | 67 |
| 13 | 59 | 30 | 84 |
| 14 | 49 | 30 | 65 |
| 15 | 53 | 36 | 68 |
| 16 | 53 | 26 | 76 |
| 17 | 53 | 21 | 79 |
| 18 | 49 | 33 | 62 |
| 19 | 55 | 56 | 55 |
| 20 | 58 | 42 | 71 |
| 21 | 55 | 51 | 59 |
| 22 | 53 | 45 | 59 |
| 23 | 47 | 38 | 54 |
| 24 | 47 | 47 | 48 |

| **HOL Ribbon Sectors** (kNN k=1) **Internal Testing** | | | |
|---|---|---|---|
| Margin | Acc. | Reg. | Irreg. |
| 10 | 60 | 58 | 62 |
| 11 | 44 | 64 | 27 |
| 12 | 44 | 64 | 27 |
| 13 | 48 | 28 | 65 |
| 14 | 50 | 32 | 65 |
| 15 | 44 | 55 | 35 |
| 16 | 54 | 77 | 35 |
| 17 | 50 | 74 | 30 |
| 18 | 52 | 93 | 17 |
| 19 | 53 | 85 | 25 |
| 20 | 47 | 81 | 19 |
| 21 | 53 | 53 | 52 |
| 22 | 58 | 32 | 79 |
| 23 | 52 | 17 | 81 |
| 24 | 48 | 40 | 56 |

## B.8 CNN-based Method

Table 12: VGG16 model using ribbon width (margin) of 12.

| VGG16 Margin=12 | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Fold | Acc. | Reg. | Irreg. | Acc | Acc | Acc | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 89 | 88 | 89 | 82 | 80 | 84 | 74 | 72 | 78 | 81 | 82 | 80 | 88 | 88 | 87 |
| 2 | 86 | 95 | 75 | 81 | 84 | 77 | 75 | 77 | 73 | 82 | 88 | 76 | 88 | 95 | 77 |
| 3 | 90 | 95 | 83 | 80 | 82 | 77 | 76 | 77 | 75 | 83 | 88 | 78 | 88 | 93 | 80 |
| 4 | 91 | 88 | 94 | 78 | 79 | 77 | 76 | 75 | 78 | 81 | 84 | 78 | 86 | 91 | 80 |
| 5 | 90 | 86 | 94 | 80 | 79 | 82 | 74 | 72 | 78 | 81 | 82 | 80 | 86 | 88 | 83 |
| **Avrg** | 89 | 91 | 87 | 80 | 81 | 80 | 75 | 74 | 76 | 82 | 85 | 78 | 87 | 91 | 81 |

Table 13: VGG16 model using ribbon width (margin) of 18.

| VGG16 Margin=18 | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| **One Doctor** | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Fold | Acc. | Reg. | Irreg. | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 85 | 84 | 86 | 81 | 79 | 84 | 75 | 72 | 80 | 84 | 84 | 84 | 88 | 91 | 83 |
| 2 | 91 | 93 | 89 | 85 | 86 | 84 | 77 | 77 | 78 | 84 | 88 | 80 | 92 | 95 | 87 |
| 3 | 91 | 95 | 86 | 84 | 80 | 89 | 76 | 72 | 83 | 85 | 84 | 86 | 90 | 91 | 90 |
| 4 | 87 | 86 | 89 | 80 | 79 | 82 | 74 | 72 | 78 | 81 | 82 | 80 | 86 | 88 | 83 |
| 5 | 90 | 86 | 94 | 81 | 80 | 82 | 73 | 72 | 75 | 80 | 82 | 78 | 86 | 88 | 83 |
| **Avrg** | 89 | 89 | 89 | 82 | 81 | 84 | 75 | 73 | 79 | 83 | 84 | 81 | 88 | 91 | 85 |

Table 14: ResNet50 model using ribbon width (margin) of 12.

| ResNet50 Margin=12 | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| **One Doctor** | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Fold | Acc. | Reg. | Irreg. | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 81 | 74 | 89 | 77 | 64 | 93 | 69 | 57 | 88 | 76 | 65 | 88 | 81 | 74 | 90 |
| 2 | 84 | 84 | 83 | 81 | 80 | 82 | 79 | 77 | 83 | 82 | 84 | 80 | 90 | 93 | 87 |
| 3 | 90 | 95 | 83 | 77 | 68 | 89 | 71 | 62 | 85 | 80 | 73 | 88 | 82 | 77 | 90 |
| 4 | 89 | 86 | 92 | 78 | 75 | 82 | 68 | 65 | 73 | 77 | 76 | 78 | 82 | 81 | 83 |
| 5 | 83 | 74 | 94 | 78 | 68 | 91 | 70 | 60 | 85 | 79 | 71 | 88 | 82 | 77 | 90 |
| **Avrg** | 85 | 83 | 88 | 78 | 71 | 87 | 71 | 64 | 83 | 79 | 74 | 84 | 84 | 80 | 88 |

Table 15: ResNet50 model using ribbon width (margin) of 18.

| ResNet50 Margin=18 | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Internal Testing** | | | | **External Testing** | | | | | | | | | | | |
| **One Doctor** | | | | **Doctor A** | | | **Doctor B** | | | **Doctor C** | | | **Agreed Cases** | | |
| Fold | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg | Acc | Reg | Irreg |
| 1 | 81 | 79 | 84 | 82 | 77 | 89 | 78 | 72 | 88 | 87 | 84 | 90 | 92 | 91 | 93 |
| 2 | 81 | 88 | 72 | 77 | 80 | 73 | 79 | 80 | 78 | 84 | 90 | 78 | 88 | 93 | 80 |
| 3 | 86 | 86 | 86 | 76 | 71 | 82 | 76 | 70 | 85 | 87 | 84 | 90 | 86 | 86 | 87 |
| 4 | 89 | 84 | 94 | 77 | 79 | 75 | 75 | 75 | 75 | 82 | 86 | 78 | 85 | 91 | 77 |
| 5 | 81 | 79 | 83 | 78 | 79 | 77 | 72 | 72 | 73 | 79 | 82 | 76 | 84 | 88 | 77 |
| **Avrg** | 84 | 83 | 84 | 78 | 77 | 79 | 76 | 74 | 80 | 84 | 85 | 82 | 87 | 90 | 83 |