Addressing Data Limitations in Defect Detection: A Case Study of Inspection in Automated Fiber Placement

by

Assef Ghamisi
B.Sc., Amirkabir University of Technology, 2020

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF APPLIED SCIENCE

in the Department of Electrical and Computer Engineering

Addressing Data Limitations in Defect Detection:

A Case Study of Inspection in Automated Fiber Placement

by

Assef Ghamisi

B.Sc., Amirkabir University of Technology, 2020

Supervisory Committee

Dr. H. Najjaran, Supervisor

(Department of Electrical and Computer Engineering)

Dr. D. Capson, Departmental Member

(Department of Electrical and Computer Engineering)

ABSTRACT

This thesis introduces novel automated visual defect detection approaches that effectively address the challenges of data scarcity and imbalance. In the manufacturing industry, conventional defect detection systems rely on end-to-end supervised learning methods that necessitate abundant labeled data, including defective samples. However, such data is often insufficiently available. In light of this, we propose two alternative approaches. The first approach combines unsupervised learning anomaly detection with rule-based computer vision, enabling effective defect detection with a smaller dataset consisting exclusively of non-defective samples. The second approach leverages rule-based computer vision exclusively, eliminating the need for any training data.

To demonstrate the practicality and efficacy of the proposed approaches, this study uses the case study of Automated Fiber Placement (AFP) and design, implement, and evaluate both methods for defect detection in this industry. Specifically, these methods are tested on depth map images of the composite surface obtained using Optical Coherence Tomography (OCT) technology. Before utilizing these images for defect detection, certain preprocessing steps, such as noise filtering, are applied to enhance their quality.

In the anomaly detection approach, the process begins with utilizing Hough Transform to estimate the boundaries of each composite strip (tow). Subsequently, a sliding window traverses along each tow, extracting small patches. A subset of these patches that are free from anomalies is used to train the autoencoder. Since the autoencoder is trained using normal samples, it can generate more precise reconstructions for these patches compared to abnormal ones. Consequently, the reconstruction error value serves as a quantitative metric to determine the presence of potential anomalies within each patch. By aggregating these values, an anomaly map is generated, en-

abling the identification of manufacturing defects within the depth map. The results demonstrate that despite the autoencoder being trained with a limited number of images, the proposed approach achieves satisfactory accuracy in binary classification and effectively localizes the defects.

The rule-based method proposed in this study effectively identifies gaps and overlaps, which are the most common manufacturing defects in AFP. This approach combines classical computer vision techniques to identify the outlines of individual tows, enabling a comparison between consecutive tows to identify any potential gaps or overlaps. To assess the effectiveness of the proposed approach, this study compares the detected defects with ground truth annotations provided by human experts. The results affirm a high accuracy in segmenting gaps and overlaps.

Table of Contents

List of Tables

List of Figures

Acronyms

1D One-dimensional. ix, 35, 36, 40, 57

2D Two-dimensional. 13, 23, 28, 35, 38, 55, 59

3D Three-dimensional. 23, 25, 59

AFP Automated Fiber Placement. iii–v, viii, 1–6, 8, 16–27, 34, 37, 42, 45, 56, 59–62

AI Artificial Intelligence. 19

AUC Area Under Curve. 53, 54

CAD Computer-aided Design. 18

CAE Convolutional Autoencoder. vii, 15–17, 38, 39, 43

CNN Convolutional Neural Network. 3, 15, 37, 38

CV Computer Vision. v, 4, 8, 11, 13, 19, 29

DoG Difference of Gaussians. 40

GAN Generative Adversarial Network. 15

HT Hough Transform. viii, 12–14

IoU Intersection over Union. 32, 48, 49, 56, 57, 60, 61

mAP mean Average Precision. 55, 56

MSE Mean Squared Error. ix, 39, 51–56

OCT Optical Coherence Tomography. iii, viii, 6, 19, 22–26, 59

R-CNN Region-based Convolutional Neural Network. 20

ROC Receiver Operating Characteristic. ix, 16, 39, 51–54

ROI Region of Interest. 4, 20, 34

VAE Variational Autoencoder. 15, 16

ACKNOWLEDGEMENTS

I would like to thank:

- My family and friends, for supporting me in the low moments.
- Dr. Najjaran and the Advanced Control and Intelligent Systems (ACIS) team, for support, encouragement, and patience.
- And the members of LlamaZOO Interactive Inc., National Research Council of Canada (NRC), and Fives Lund LLC, for their research collaboration on an industrial project that enabled this research work.

Chapter 1

Introduction

This chapter provides an introductory background on defect detection and Automated Fiber Placement to establish the motivation for this thesis. The research questions and objectives of the thesis are stated, and the contributions of this work are discussed. Finally, the chapter concludes with an overview of the thesis organization.

1.1 Background and motivation

Automated Fiber Placement (AFP) is an advanced manufacturing technique that produces composite materials. It involves the precise robotic placement of continuous fiber tapes onto a surface to create strong, lightweight structures. The composite parts manufactured using AFP find applications in industries that prioritize high quality and reliability such as aerospace and automotive [1, 2, 3].

Similar to any manufacturing process, AFP is susceptible to defects, which can have a substantial impact on the mechanical properties of the composite part [4, 1]. For instance, the existence of systematic defects can decrease the strength of the composite structure [5]. Consequently, inspection plays a critical role in AFP by identifying defects and ensuring the production of defect-free composite parts.

Defect detection in AFP has traditionally relied on manual inspection, where human experts visually examine each strip of composite fiber before proceeding to the next layers [6]. However, this manual inspection process can often become time-consuming, occasionally exceeding the duration of the AFP deposition itself [7]. Consequently, the accuracy and efficiency of manual inspection are not consistently reliable.

To address these challenges, researchers have focused on automating defect detection in AFP. Automated inspection technologies have been developed and implemented within AFP processes to overcome the limitations of manual inspection. These advanced technologies utilize computer vision, image processing, and machine learning algorithms for real-time defect detection and classification. By leveraging these automated systems, the detection of defects becomes more efficient and accurate in AFP manufacturing.

This thesis aims to explore and evaluate the effectiveness of automated defect detection systems, specifically focusing on computer vision techniques, in the context of Automated Fiber Placement (AFP). By investigating the capabilities and limitations of computer vision-based defect detection in AFP processes, this research contributes to improving quality control in composite manufacturing. Furthermore, this study introduces novel approaches for defect detection in AFP, incorporating innovative methods that have not been extensively explored in previous research. Integrating these novel approaches enhances the potential for accurate and efficient defect detection, paving the way for advancements in the field. The findings of this study can provide valuable insights and recommendations for industry practitioners seeking to enhance the efficiency and reliability of defect detection in AFP, ultimately leading to improved product quality and reduced production costs.

1.2 Research Question and Objectives

Problems: Data Scarcity and Imbalance

Most of the existing defect detection solutions heavily rely on end-to-end supervised learning, utilizing Convolutional Neural Networks (CNNs) to train on labeled datasets. However, these supervised learning methods can be impractical for industrial projects due to the specific requirements associated with training data.

Firstly, the abundance of training data poses a challenge. End-to-end learning necessitates a large number of images of manufactured parts, which may not always be readily available.

Secondly, data labeling is time-consuming and manual [8], adding further complications to supervised learning. Additionally, there is no universally accepted standard for how a human inspector should label defects for machine-learning purposes [7].

Thirdly, defects in real-world production are infrequent [9], resulting in a severe data imbalance issue. Normal samples are abundant, while defect samples are scarce [8]. Manually inducing manufacturing defects for data generation is expensive, impractical, and can lead to destructive consequences. This inherent data imbalance problem greatly complicates the effectiveness of supervised learning approaches [9].

To tackle these challenges, researchers have proposed various solutions to address the data imbalance and scarcity issues [10]. This study presents two alternative approaches to overcome the limitations of supervised learning for defect detection.

Solution 1: Unsupervised Anomaly Detection

Unsupervised learning is one of the solutions to tackle the data imbalance problem [8]. In AFP, the non-defective parts with no defects are typically well-defined thanks to the simple, invariant structure of the tows. Non-defective samples are used, which

constitute the major proportion of inspection data, to train a classifier to discern abnormal samples that appear different from its training set. This work applies the Autoencoder methodology in AFP anomaly detection, thanks to its robustness and efficiency in creating an anomaly classifier that only requires a limited size of normal samples. An autoencoder works by learning to encode normal input samples into a lower dimensional latent vector that can be decoded to reconstruct the original sample. The reconstruction is compared with the original, and a reconstruction error metric is calculated. When an abnormal sample is provided, the reconstruction errors will be high since the autoencoder was never trained with a similar image. A threshold is then applied to the reconstruction errors to determine if the sample is normal or abnormal.

The main drawback of this approach is that it only determines the anomalies and does not identify the defect type. On the other hand, it offers several advantages over the existing methods:

- It requires fewer samples by using efficient Region of Interest (ROI) selection.
- Labeling is not necessary; the sole requirement is to ensure the absence of defects in the data.
- It only needs non-defective samples, which is the majority of samples.
- It only relies on the understanding of normal samples, which is well-defined compared to poor definition of defect.

Solution 2: Rule-based Computer Vision

Clasical CV has popularity for object detection due to its efficiency and reliability. More importantly, these methods do not necessitate the training of any machine learning model. This fact solves the data scarcity and imbalance problems in defect

Method	Data required	ML-based	Rule-base
End-to-end supervised learning	Abundant labelled data including defective samples	✓	
Unsupervised anomaly detection with efficient sample extraction	Fewer data of inclusively defective samples with no labeling	✓	✓
Gap and overlap segmentation using curve detection	None		✓

Table 1.1: Comparison of the existing and proposed defect detection methods

detection. This study proposes a novel approach to the gap and overlap detection in AFP by leveraging the composite tapes (tows) known structure to create a customized tool to segment gaps and overlaps in composite parts. This approach is evaluated on samples of composite profilometry images to compare its performance with existing methods. The experimental results demonstrate that this approach can achieve high accuracy in the gap and overlap segmentation, making it a promising tool for improving the efficiency and quality of the AFP process.

Table 1.1 summarized the comparison between the common existing and proposed methods in this work.

1.3 Contributions

Contributions of this research are as follows.

1. Presenting a literature review on the existing defect detection techniques, especially in Automated Fiber Placement.

2. Introducing a novel framework for anomaly detection and localization in Automated Fiber Placement. Using the preliminary knowledge of the compos-

ite tows' symmetry, the proposed approach only requires a small dataset of non-defective composite parts, which overcomes the problem of defective data scarcity.

3. Designing and validating an autoencoder with the optimal size of the latent domain that can identify the best distinctive features to differentiate between normal and defective samples.

4. Presenting a novel rule-based approach for the detection and demonstration of gaps and overlaps in the process of Automated Fiber Placement which does not require any data for training.

5. Implementing the presented methods to prove their feasibility.

6. Testing the impact of individual components of gap and overlap detection workflow to show the significance of each step.

7. Evaluating the performance of the defect detection methods by comparing the results with human-annotated ground truth images.

In this thesis, OCT sensor data is used to present the inspection techniques. However, these solutions can be scaled to other AFP imaging techniques with some modifications.

1.4 Organization

The organization of this thesis is as follows:

Chapter 1 presents an introduction on the thesis.

Chapter 2 provides a literature review on defect detection and Automated Fiber Placement.

Chapter 3 describes, the full procedure of anomaly detection, including data preprocessing, training of the AI model, and implementation details.

Chapter 4 describes the experimental setup and explains the implementation details.

Chapter 5 presents the evaluation results of both the anomaly detector and the localization system. Also, the impact of various components in the algorithm is evaluated.

Chapter 6 concludes the thesis and suggests future research directions.

Chapter 2

Literature Review

This chapter provides an overview of the classical and learning-based computer vision techniques employed in this thesis. These techniques play a crucial role in tasks such as image preprocessing, visual information extraction, and object detection. Furthermore, a comprehensive examination of the existing literature on defect and anomaly detection is presented. Lastly, an analysis of defects, inspection technologies, and methods in Automated Fiber Placement (AFP) is included in this review.

2.1 Classical Techniques in Computer Vision

Median Filters for Noise Reduction

Image de-noising is a common image pre-processing step. This step is very important as it generates a clearer representation of images, as a noisy image can significantly damage the following image processing and computer vision tasks. Image filters, including linear and non-linear ones, have been used to remove or reduce image noise. Linear filters result in a blurring effect on the image details like edges, whereas non-linear filters can preserve these details while reducing the impact of noise [11].

One of the most common noise types is salt and pepper noise. Images can have random impulse noise due to hardware (e.g., camera sensor) or software (e.g., memory allocation) problems [12]. In salt and pepper noise, random values are either minimum (black) or maximum (white) values [13]. Nonlinear filters, especially median ones, are commonly used to remove salt and pepper noise [11, 12, 14].

A median filter substitutes a pixel's value with the median of the neighboring pixels within its vicinity. A mask defines this neighborhood and often includes the original pixel [15]. Different mask shapes can be applied, such as squares, circles, crosses, etc. [14]. A mask is moved over the image. For each position of the mask, the image pixels placed within the mask are gathered and sorted in ascending order. The median value of these numbers is used as the filtered value of the center pixel in the resulting image [13]. Justusson (1981) provides a formulation of the median filter as Equation 2.1 [14]. In this equation, a median filter (med) is applied to image x and generates image y as the filtered result. The median of the values is only calculated for the pixels masked by the filter A.

$$y_{ij} = \underset{A}{\text{med}} x_{ij} \triangleq \text{med}\left[x_{i+r,j+s}; (r,s) \in A\right], \quad (i,j) \in \mathbb{Z}^2 \tag{2.1}$$

In Figure 2.1, the effect of the median filter on reducing the impulse noise on an image is demonstrated. The outliers of the image are removed while the visual details are preserved.

Morphological Operations

Morphological operations are essential image-processing techniques used to analyze and modify the shape and structure of objects in an image. Morphology has been used in different image-processing tasks [16]. For example, Li et al. (2015) used Morphological operations to de-noise the input images to detect bubbles in blood

(a) Image with salt and pepper noise (b) Image after median filtering

Figure 2.1: The impact of median filter on removing salt and pepper noise [13].

dialysis devices [17].

These operations are executed by utilizing a structuring element, a small shape or pattern that defines the surrounding area of each pixel. The structuring element is usually a binary matrix with a predetermined shape, such as a square, circle, or costume shape. Selecting a proper structuring element is important as it changes the opening and closing output [18].

Dilation and erosion form the basics of morphological operations. Dilation expands the objects in the image while erosion shrinks them [17]. By combining these two operations, more complex operations can be achieved. For example, the opening is a combination of erosion followed by dilation and is used to remove noise and small objects from an image. Conversely, closing involves dilation followed by erosion and is useful for filling gaps and holes in objects. The opening operation eliminates small groups of pixels and narrow lines that belong to an object [19]. Therefore, it is used for image filtering [20]. Equation 2.2 shows the opening operation ($\circ$) in which $\ominus$ and $\oplus$ are erosion and dilation operations, respectively. The image (I) opened using the structuring element (S) is defined as first eroding I with S and then dilating

the resulting image with S [17]. Note that the same structuring element is used for erosion and dilation [16].

$$I \circ S = (I \ominus S) \oplus S \tag{2.2}$$

Figure 2.2 demonstrates the impact of four basic morphological operations on a binary image.

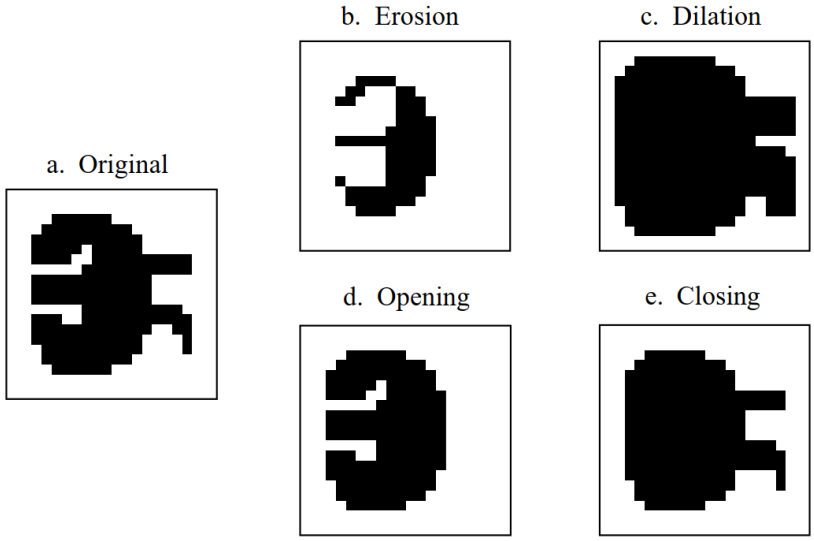


Figure 2.2: Morphological Operations [19]: object shown in black and background in white pixels

Edge Detection

In Computer Vision, an edge is a discontinuity in the image intensity [21]. Edges represent significant visual information [22], like the boundaries of the objects with different properties. Edge detection is the process of finding sudden changes (sharp transitions) in the image intensity. finding the edges extracts invaluable information on the shape and boundaries of the objects in an image. Edge detection is a significant technique in image processing and computer vision. It can be used to condense the information in an image by eliminating unnecessary data while retaining the essential

details [21]. Edge detection has been used for various computer vision tasks such as segmentation, object detection, image enhancement, etc. [23]. For example, one of the edge detection applications is finding the objects' boundary in an image [24]. However, edge detection is not a trivial task. For instance, edge detection in a noisy image is challenging because the high-frequency nature of noise can be misinterpreted as rapid intensity changes of edges [21].

Different methods have been used for edge detection. Sobel [25] is one of the fundamental edge detectors. This operator is useful for finding the edges along a certain direction [21]. Sobel edge detector uses two directional kernels to find the horizontal and vertical gradients of the image. By combining these two edge components, the magnitude, and direction of the edges are calculated. After calculating the gradient, the gradient values above a threshold represent edges. Finding the best threshold is one of the most challenging parts of Sobel edge detector [24].

A more sophisticated method is the Canny edge detector [26], among the most popular edge detection techniques in Computer Vision [24]. Canny edge detection uses the basics of Sobel, but it has extra steps of non-maximum suppression and hysteresis [24] that make it a more powerful tool. In many cases, Canny edge detection outperforms Sobel edge detection [27], especially for the detection of long edges [24]. This edge detector is robust to noise as it uses image smoothing to remove the effect of noise [21].

Hough Transform for Line Detection

Edge detection can extract valuable visual information on the object's shape, but it usually generates a discontinuous representation, which is not always enough to detect an object. Various methods have been used to generate the original representation of the shapes, like lines. One of the most powerful techniques is Hough Transform

(HT) [28], which is a voting-based feature extraction technique. HT, simplifies the complicated problem of finding objects in image space to finding local maxima in parameter space [29]. HT was first used for detecting lines [28] but was extended to more shapes [30]. This approach is robust to noise and partial occlusion and works in the presence of other unwanted objects in the image [29].

Two types of Hough Transform are Classical and Generalized. Classical HT requires an analytical shape expressible with a few parameters [31]. For example, lines can be represented with two and circles with three parameters. Generalized Hough Transform is useful for detecting arbitrary shapes and curves and is used when no analytical shape representation is available. However, this approach is more computationally expensive.

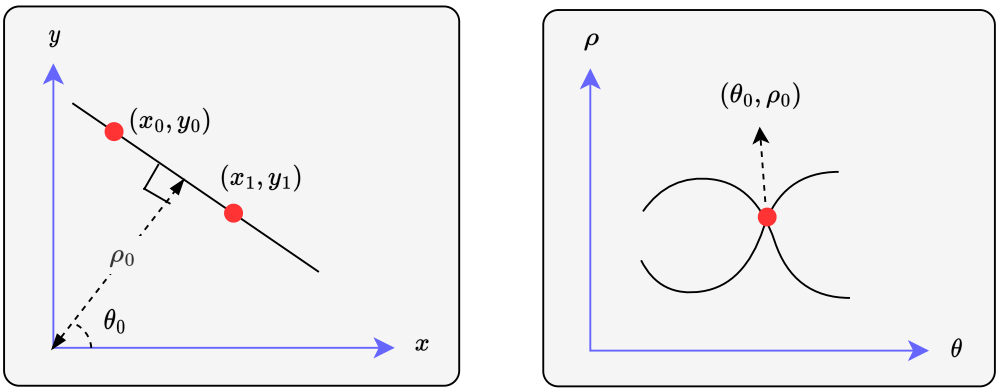
Hough Transform involves multiple steps. First, the edges of an image are extracted using an edge detector (e.g., Canny). Then each edge point is transferred from Cartesian pixel space into a parametric space known as Hough space, generating an accumulation of votes for the object. The local maxima in Hough space are found as candidates for the desired objects. Finally, the most prominent objects are detected using post-processing techniques like thresholding.

Lines are among the most valuable shapes in Computer Vision. Finding lines is helpful in various applications, such as lane detection in autonomous driving. Hough Transform has been a popular method for line detection [31]. Only two parameters are enough to represent lines in a 2D space. In polar coordinates, a line with distance ρ from the origin and angle of normal to the line θ is represented in 2.3 [30]. A sample line is shown in Figure 2.3a.

$$\rho = x\ cos(\theta) + y\ sin(\theta) \tag{2.3}$$

All the possible lines parameterized as (θ, ρ) passing the edge pixel at (x_0, y_0) are

calculated and accumulated to the Hough space (Figure 2.3b).



(a) A line in Polar Coordinates (b) Parametric Hough Space

Figure 2.3: Transfering a line from image space into Hough Transform space

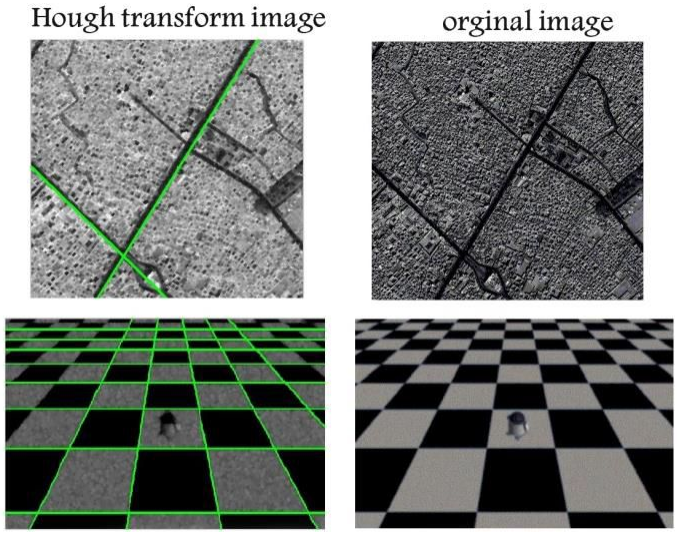


Figure 2.4: Hough transform is used to detect the most prominent lines in grayscale images [32]

2.2 Anomaly Detection

Manufacturing defects can be considered as anomalies. An anomaly is defined as any unusual observation in a dataset.

There are several methods to detect anomalies in images including supervised, semi-supervised, and unsupervised methods [33]. Supervised anomaly detectors require a large, labeled dataset to learn the binary classification of normal or anomaly. Semi-supervised methods also required labeled data, however, these approaches assume that only non-anomalous samples are present in the training dataset. The semi-supervised methods generally intend to learn a compact mapping or encoding that represents the normal data well, while poorly representing anomalies [34]. Classification is then performed based on a selected metric, usually a reconstruction error, to identify samples that are not well-represented by the learned model. Unsupervised methods, on the other hand, have no label requirements for the data and attempt to learn high-level representations or distributions of the data to identify outliers as anomalies [35]. The state-of-the-art image-based anomaly detectors mainly employ Convolutional Neural Networks (CNNs) with semi-supervised and unsupervised methods, using generative models such as Autoencoders and Generative Adversarial Networks (GANs [36]).

Autoencoders, with the ability to reconstruct input samples, are powerful tools for classifying anomalies in images. Ulger et al. (2021) [37] investigate the use of autoencoders to detect defects in solder joints. The authors compare three different types of autoencoders as anomaly detectors to identify the defects: a simple Convolutional Autoencoder, a Variational Autoencoder, and a Disentangled Variational Autoencoder. The autoencoders learn to encode non-defective input samples into a lower dimensional latent vector that can be decoded to reconstruct the original sample. With this, the reconstruction is compared with the original, calculating a reconstruction error metric. A threshold is applied to the reconstruction errors to determine if the sample is similar to the distribution of the training dataset. If the reconstruction error is too high, the sample is classified as a defect. The implemented

autoencoders all follow a similar convolutional architecture, however, the variation autoencoder uses Gaussian distributions to represent the encoded samples. The disentangled variational autoencoder is quite similar, with an additional term added to the loss function that incentivizes the model to learn less-correlated distributions. This method could be suitable for defect detection in AFP composites as the model only requires non-defective samples for training, which are much more abundant than defect cases.

With a similar approach for detecting surface defects of manufactured materials, Tsai et al. (2021) [38] investigate Convolutional Autoencoders (CAEs) for anomaly detection based on reconstruction error. The autoencoder models are used to extract representative features for image inspection in contrast of traditional machine learning methods that require handcrafted feature extractors. The proposed anomaly detector is tested on various different textured and patterned surface types, including wood, liquid crystal displays, and fibreglass. This work also implements a VAE with a similar architecture to the CAE for comparison. Both the CAE and VAE models use 320-dimensional latent vectors to represent the encoded image data. Classification is conducted based on the reconstruction error, with a threshold determined by a Receiver Operating Characteristic (ROC) curve. Comparing the ROC curves of the CAE and the VAE, the authors found that the CAE could achieve much better classification thresholding than the VAE. Overall the results indicate good classification performance and efficient processing. The authors note that their method is an image-wise defect detection approach, however, it could be adapted to localize defects by sliding a window over the image to classify local regions. This approach could be very beneficial for anomaly detection in industrial inspection industries where data of defective samples are very limited.

In the same way, Chow et al. (2020) [39] implement a reconstruction-based

anomaly detector using a Convolutional Autoencoder to detect surface defects on concrete structures. They use a semi-supervised approach, training the autoencoder with a dataset of defect-free image samples. Their model learns to minimize the reconstruction error of non-defective inputs but produces high reconstruction errors for defective inputs. However, the input data for concrete structure inspection are generally high-resolution images that cannot be processed all at once due to the limitations of computation resources. To solve this issue, the authors implement a window-based approach that takes smaller cropped sections of the overall image as input to the network. This is done by sliding a window along the image at different scales and rescaling the image patches to a standard input size of the network. This also enables the model to better detect anomalies of various sizes. In this approach, the combined reconstruction errors are used to generate a pixel-wise anomaly map for each full-size image. The anomaly map produces a score for each pixel, rather than just a binary classification, which is used as an indication for inspectors to understand the risk level associated with the anomalies. The results of this process show improved segmentation performance when compared with other classical methods and even add contextual understanding with the pixel-wise scoring.

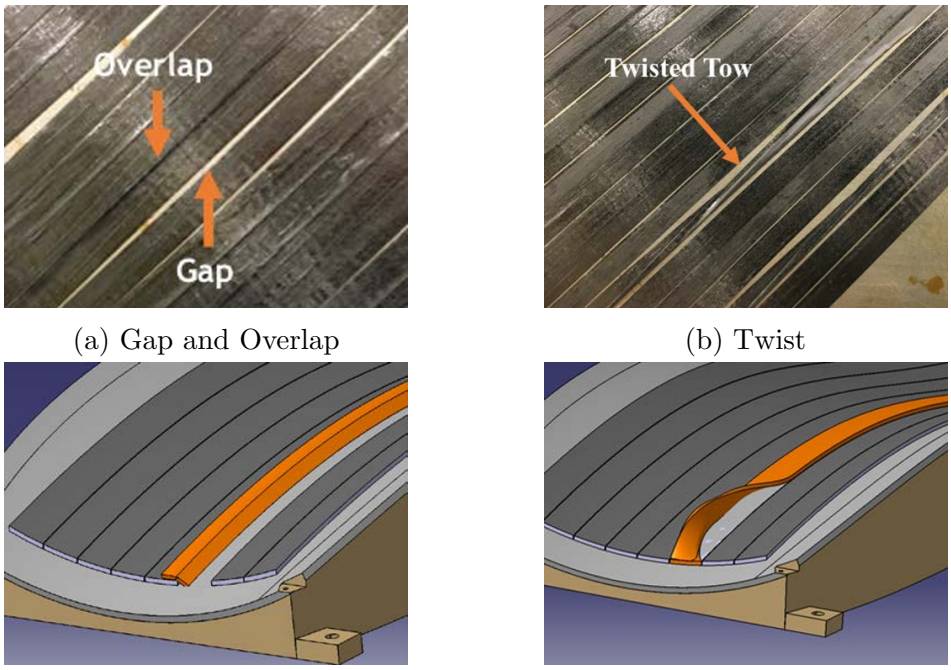
2.3 Automated Fiber Placement

Defects

In 2019, Heinecke and Willberg reviewed manufacturing-induced defects in AFP, identifying the types of defects in this industry and analyzing different methods for evaluating the effects of each defect [7]. In their paper, defects are classified into 10 different categories based on how they occur and their resulting effects. The defect types identified are: angle deviation; tow misalignment; waviness; gap and overlap;

twisted tow; bridging and crowning; tow wrinkling and upfolding; voidage and inclusions; residual stresses and process-induced deformations; and fuzz formation. The authors note that defects in this industry have a very low likelihood of occurrence and can be challenging to reproduce. For this reason, many of the identified defects are rarely investigated. In their review, Heinecke and Willberg identify the lack of well-developed monitoring systems for AFP production processes.

Also in 2018, Harik et al. [40], reviewed the defects in AFP. In Figure 2.5 two Computer-aided Design (CAD) models for two types of defects are illustrated. The defects can be either between the tows (like gaps and overlaps) or within a tow (like twist).



(a) Gap and Overlap

(b) Twist

(c) Gap and Overlap

(d) Twist

Figure 2.5: Some common types of AFP defects are shown as photos (top row) and CAD models (bottom row) [40].

Gaps and overlaps are the most frequent defects in AFP [7, 40, 41]. A gap happens when two neighboring tows are misaligned in a way that there is some space between them. On the other hand, an overlap occurs when the two adjacent tows are positioned

such that they partially cover each other [40]. The presence of gaps and overlaps can negatively impact the mechanical properties of the final part. They may results in uneven distribution of thickness [41], and strength reduction [42] in the laminate. Also, they can form regions within the composite part that are excessively rich in either resin or fiber [43].

Inspection Techniques

Many non-destructive testing methods have been used to evaluate composite structures [44]. Some of these methods utilize profilometry to capture surface information from composite parts and present it in the form of images. Various imaging tools employ different sensors, such as optical cameras [45], thermal cameras [46, 47], and laser scanners [48, 49]. Researchers at the National Research Canada (NRC) have developed a precise imaging tool using an Optical Coherence Tomography (OCT) sensor [50].

Defect Detection Methods

Recent research seeks to automate anomaly and defect detection by using Artificial Intelligence (AI), Computer Vision (CV), and deep learning methodologies, reducing manual effort in AFP inspection, and expediting production.

There are many publications that explore supervised learning methods for inspection in manufacturing industries, including AFP.

To begin with, Sacco et al. (2020) [51] review the applications of machine learning in composite manufacturing processes and present a case study of state-of-the-art inspection software for AFP processes. The presented inspection method uses a deep convolutional neural network for semantic segmentation to classify defects on a per-pixel basis. They use about 800 scans which is relatively a large dataset in this

domain, yet the results show their method often misses some defects. One downside of semantic or instance segmentation is that it requires pixel-wise data labelling. In an industry project, however, it is not common to have pixel-wise annotated data or in practice, it is very time-consuming to create.

Another approach using deep learning to classify defects in AFP is presented by Schmidt et al. (2019) [52]. The work employs convolutional neural networks to analyze thermal images of composite materials taken during the AFP process. Their detection and classification method is broken down into two stages. The first stage monitors each tow individually, detecting and classifying defects. In the second stage, the total width of faultless tows is monitored, while multi-tow defects are detected and classified. The models are trained on a dataset consisting of 1000 images per class.

In a similar problem domain, Wei et al. (2019) [53] present a defect detection method for fabric and textiles based on the faster Region-based Convolutional Neural Network method (Faster R-CNN [54]). The dataset in this study has 6 classes of surface defects, quite similar to the defects found in AFP-manufactured composites. The modified R-CNN network consists of convolutional layers with a regional proposal network to generate anchor points for proposed areas of interest. The anchor points are passed through a Region of Interest (ROI) pooling layer and then fully connected layers to predict a bounding box and classify the defect.

Another defect detection method based on R-CNN is presented by Wang et al. (2018) [55] to identify defects on styrofoam containers. Their work implements Mask R-CNN which is a modification of R-CNN that uses an ROI Align layer in place of the ROI pooling layer. Mask R-CNN also performs pixel-based segmentation in addition to bounding box detection. The authors of this paper note that defects are typically rare and it can be difficult to compile a dataset for. In their work they

suggest that models can be trained on various different products to detect defects in a more generalized manner.

The problem with these supervised learning approaches is that their performance is heavily dependent on the dataset. Training a supervised learning classifier requires an adequate amount of defect data, which is not easily obtained. A dataset with a large number of non-defective samples and only a few samples of defects is unbalanced, and training a classifier with such a dataset increases the chance of the classifier being biased towards the non-defective class. Moreover, adding manual defects in order to generate a training set for supervised learning methods does not fully reflect the features of machine-induced defects. In consequence, the supervised methods are not practical or reliable for real inspection systems in AFP.

Chapter 3

Methodology

This chapter offers an overview of the defect detection approaches proposed in this study. It discusses the utilization of rule-based computer vision methods, such as edge detection, in the design of a gap and overlap detection system. Additionally, it explores the integration of classical computer vision techniques with unsupervised deep learning to develop an anomaly detection tool for Automated Fiber Placement (AFP). A visual representation of our methods is provided in Figure 3.1, summarizing the key steps.

3.1 Data Acquisition and Preprocessing

Data Collection

The defect detection methods employed in this study utilize data collected from the Automated Fiber Placement (AFP) setup depicted in Fig. 3.2a. Scans are conducted on composite carbon fiber surfaces using an Optical Coherence Tomography (OCT) sensor installed on the AFP layup head. The OCT sensor captures high-resolution point clouds of the layup tow surface [56, 57]. Fig. 3.2b showcases an image of the

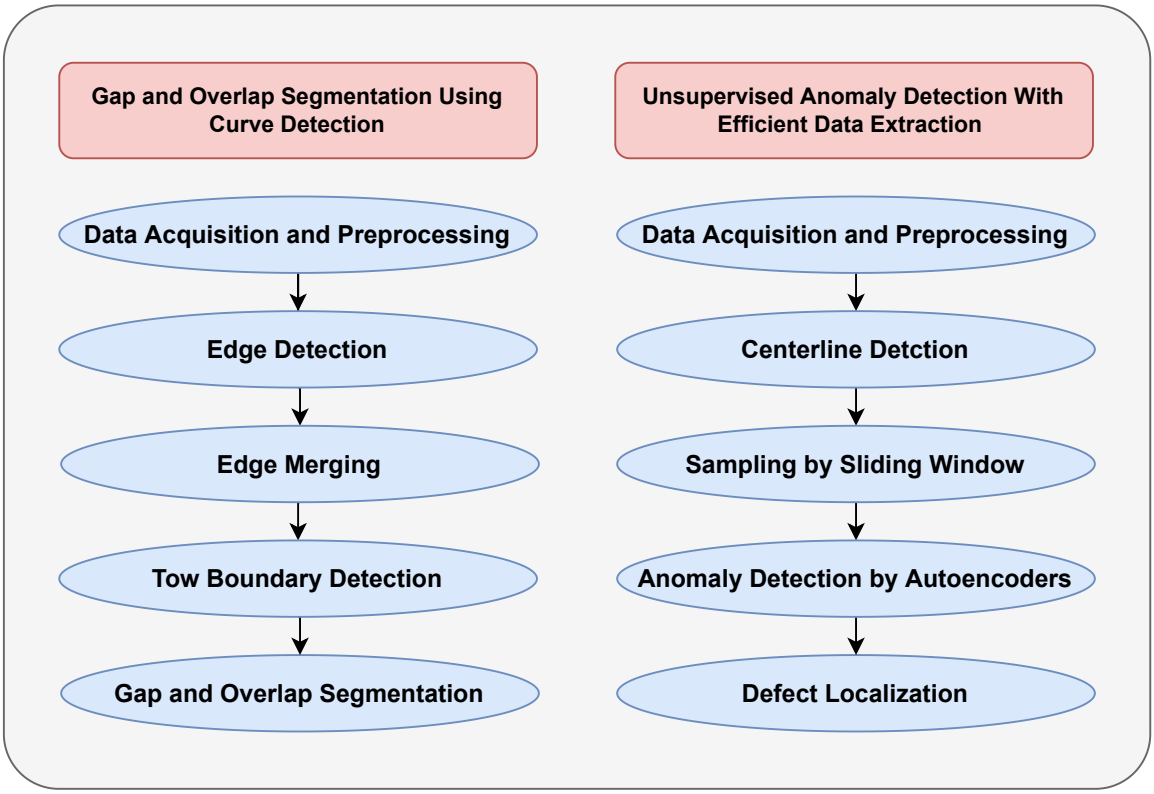
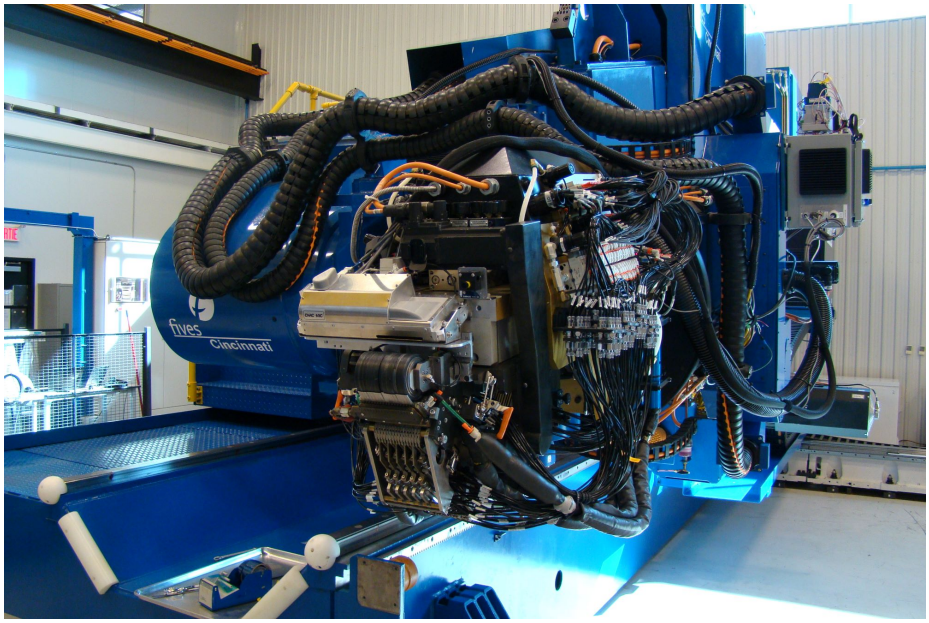


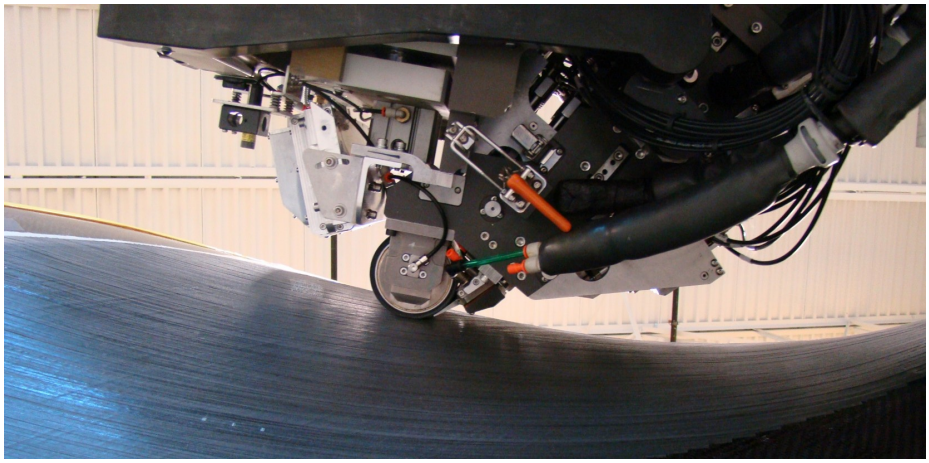
Figure 3.1: An overview of the proposed approaches.

AFP head with the OCT sensor installed.

To simplify and enhance efficiency in processing information, the 3D point clouds are converted into 2D depth maps. During this conversion, the key structural information of composite tapes is retained. These depth maps are presented as grayscale images, where the brightness of each pixel corresponds to the surface elevation on the composite part. Moreover, the location of each pixel corresponds to the Cartesian coordinates of the corresponding point on the composite part's surface. Fig. 3.3 provides various representations of a sample composite part, highlighting that defects are less discernible in the RGB image due to reflections and low visual contrast. However, defects become more evident in the depth map, offering improved visibility for defect detection.

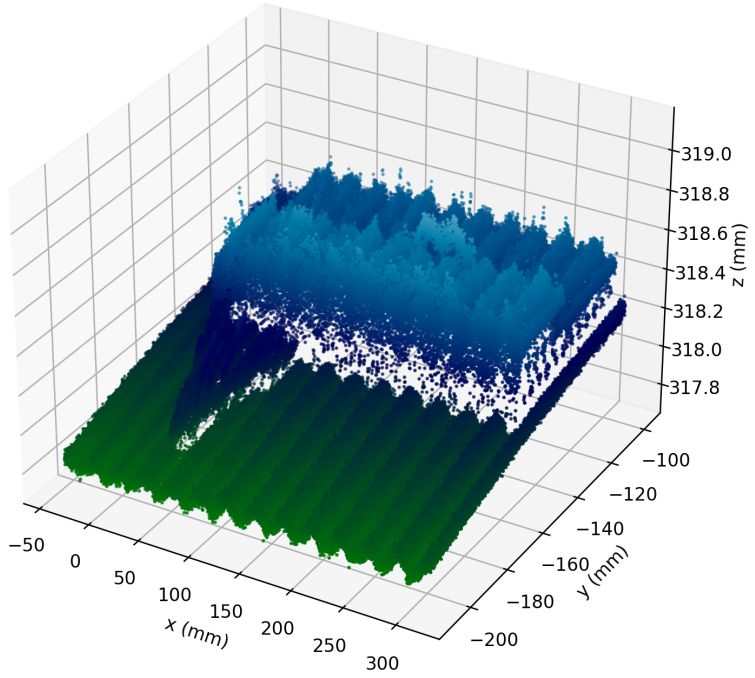


(a)

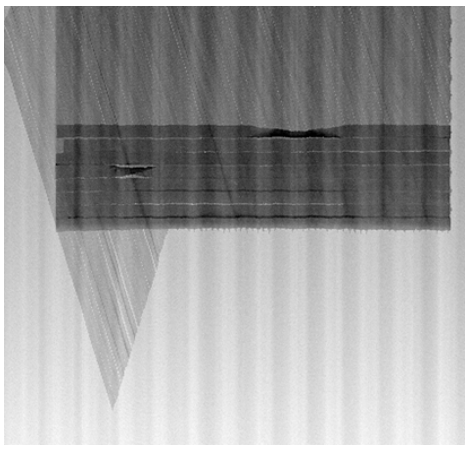


(b)

Figure 3.2: The industrial AFP setup is shown in two separate views. On the left is an overall view of the fibre placement machine (a), and on the right is a close-up shot of the robotic tool applying carbon fibre tows. The OCT sensor is visible to the upper left of the roller.



(a)



(b)

(c)

Figure 3.3: Different representations of a composite part manufactured with an AFP machine. Above (a) is a 3D point cloud measured using OCT Technology. The bottom left (b) shows the depth map generated from the 3D point cloud, and a real photograph of the same composite part is shown in the bottom right (c) for comparison.

Noise Removal

The raw depth maps contain impulse artifacts, also known as salt and pepper noise, which can be detrimental. To remove this type of noise, a median filter applies on to the depth image [58]. Compared to a Gaussian filter with a small radius (a low-pass filter) [59], a median filter has a lower risk of losing high-frequency features.

Global Normalization

Another data preparation step is needed because different raw depth-map images have inconsistent ranges of values depending on the distance of the laser origin to the composite surface. This effect is commonly caused when the OCT sensor is mounted to a fixed location behind the AFP head while scanning a contoured surface. These variations can cause undesired behavior in the defect detection methodology. To address this, all the images undergo a min-max normalization so that the minimum depth value is mapped to 0 and the maximum value is mapped to 1. By applying this linear transformation, the visual contrast of the images is improved while keeping the original depth ratio. The normalization function is provided in Eq. 3.1 in which $z_{i,j}$ is the original depth value, $p_{i,j}$ is the normalized pixel value, and Z is the whole depth map matrix.

$$p_{i,j} = \frac{z_{i,j} - \min(Z)}{\max(Z) - \min(Z)} \tag{3.1}$$

3.2 Gap and Overlap Segmentation

In this section, we explore the procedures involved in our gap and overlap segmentation algorithm. Initially, we employ capture scans from composite structures and apply fundamental image processing techniques to enhance the input image, eliminate unwanted noise. Then, we extract the edges on the image and filter only the

horizontal edges that correspond to the upper and lower boundaries of the tows. Subsequently, we proceed to group and merge the extracted edges to create continuous horizontal edges for the tows. Following that, we apply a curve-fitting approach to estimate and represent the boundaries of the tows using the merged edges. Lastly, we utilize the tow boundaries to segment any gaps and overlaps in the image.

Edge Detection and Filtering

As discussed in Chapter 2, edges in grayscale images are sharp transitions in intensity that mark the boundaries between different regions within the image. In AFP images, some edges may flag the boundaries of the tows which are useful for gap and overlap detection. To extract the edges of the image, a canny edge detection with a proper Gaussian filter size applies on the processed image. While this process captures all edges, some of them may not correspond to the upper or lower boundaries of the tows. Since our objective is to isolate only the upper and lower boundaries, it becomes necessary to eliminate any additional information or noise present in these edges. To accomplish this, an opening morphological operation is used. The opening operation uses a horizontal line with a length of 5 pixels as the structuring element which filters horizontal edge pieces. Figure 3.4 shows the impact of the opening operation on filtering out noise and non-horizontal edges.

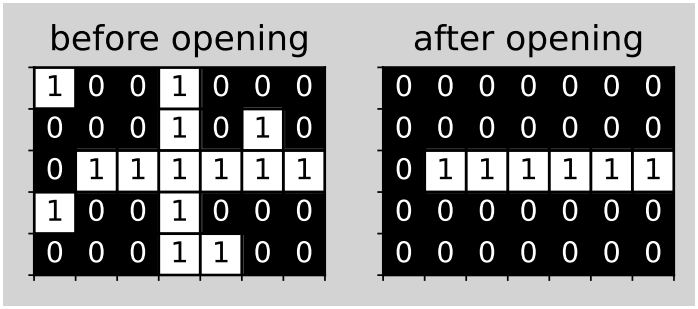
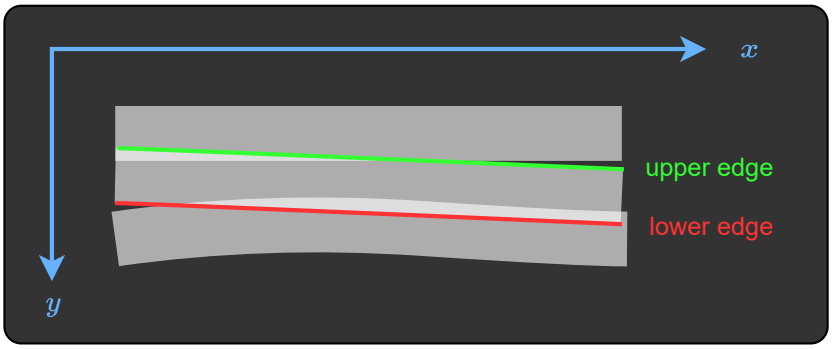


Figure 3.4: A morphological opening operation with a horizontal line structuring element only preserves the horizontal edge pieces.

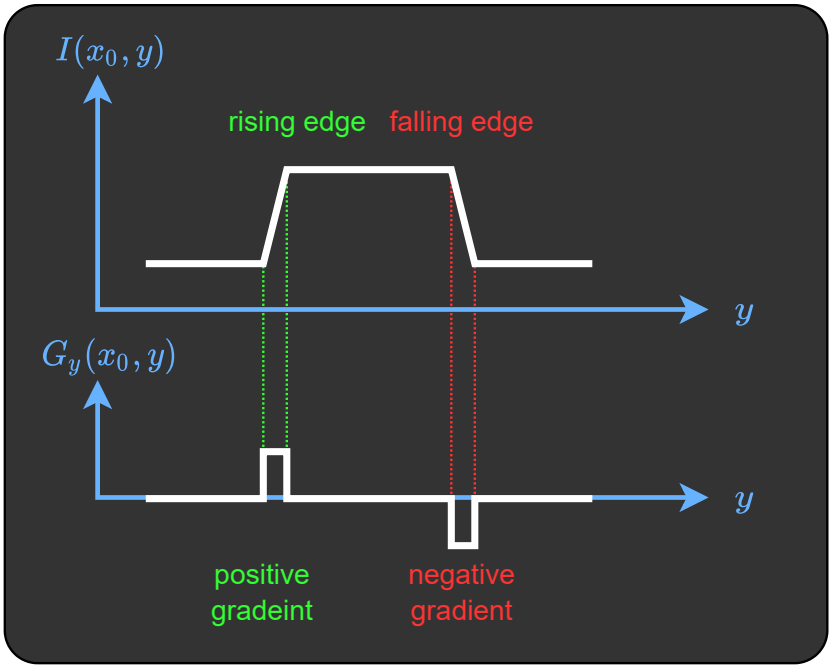
After extracting all the horizontal edges from the image, the next step is to classify these edge pieces into two distinct categories depending on whether they correspond to the upper or lower boundaries of the tows (Figure 3.5a). The challenge in this task is that the lower edge of the upper tow and the upper edge of the lower tow are often very close to each other, and sometimes they overlap. To tackle this problem, the sign of the vertical gradient of the image is used as the classification criteria. Sobel is a standard way to generate image gradients [24]. To begin with, a Sobel operator formulated as in Equation 3.2 calculates the vertical gradient denoted as G_y. Here, I represents the input image, and $*$ signifies a 2D convolution.

$$G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * I \tag{3.2}$$

Considering the up-to-down direction for the vertical axis (rows of the image) in Figure 3.5b, a rising edge in the pixel brightness values corresponds to the upper edges of a tow. As a result, positive values of G_y indicate the presence of upper boundaries. Conversely, negative values signify lower boundaries of the tows. Thus, by examining the values of the vertical gradient, the extracted horizontal edges can be effectively classified as either upper (positive) edges or lower (negative) edges.



(a) Upper and Lower Boundaries of a Tow



(b) Presence of a tow cause vertical changes in image intensity and gradient sign.

Figure 3.5: Distinction Between Upper and Lower Tow Edges

Merging the Edges

The identified horizontal edges are discontinuous, and as a result, they do not show the upper and lower boundaries of the tows. To address this, merging the edges associated with the same tow is necessary, effectively creating continuous boundaries.

First, a Computer Vision algorithm called Connected Component Labeling [60] to identifies each connected component (region) of the binary image as illustrated in

Figure 3.6. In this project, each region is a piece of the horizontal edge of the tow that is not connected to other edges.

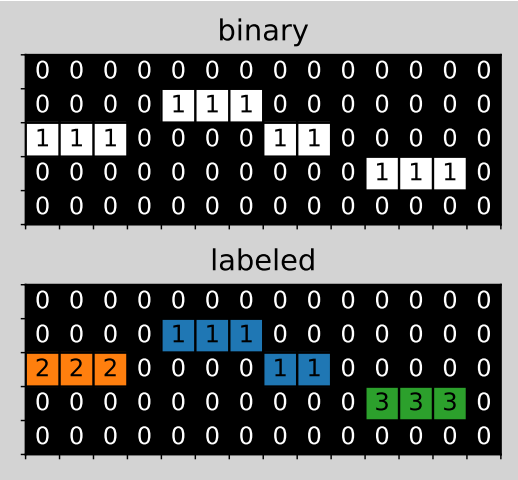


Figure 3.6: Connected components labeling of edge regions.

The edge regions that belong to the same tow should be accurately grouped to prevent a wrong connection between two regions in different tows. This edge grouping is done by defining an equivalency relation (R) provides in 3.3. The regions a and b are considered to be in the same class (tow) if their distance (d) is lower than the threshold value d_{Th}.

$$R = \{(a, b) : d(a, b) < d_{Th}\} \tag{3.3}$$

More importantly, they need to be vertically close to each other. Since vertical distance is more effective than horizontal distance, a customized distance function is employed that considers both horizontal and vertical distances of the regions with more penalty on vertical distance. This distance is formulated as equation 3.4. In this equation, d is the customized distance, Δx and Δy are horizontal and vertical distances, respectively. Also, a_1 and a_2 are constant coefficients.

The customized distance between each pair of regions is calculated. If two regions

are close, they are considered in the same equivalence class. This algorithm groups all the regions into some classes, each class representing a unique tow in the image.

$$d(a,b) = \alpha_x|x_b - x_a| + \alpha_b|y_b - y_a| \quad , \quad \alpha_x < \alpha_y \tag{3.4}$$

The next step is to merge the regions that belong to the same tow. To achieve this, a straight line connects region pairs that are horizontally next to each other. This line connects the rightmost pixel of the left region to the leftmost pixel of the right region. The process of merging these edges to form cohesive boundaries of the tows is visually depicted in Figure 3.7.

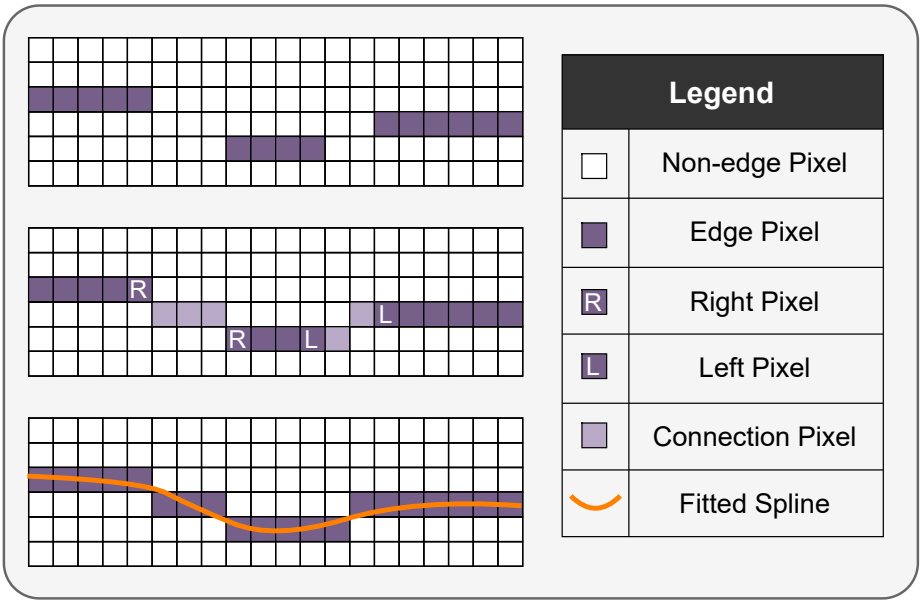


Figure 3.7: Connecting the edge regions within a tow

To obtain a clear representation of reconstructed tow boundaries, the proposed algorithm transforms the merged tow edges (pixels) into smooth curves (vectors) for visualization. This is accomplished by applying a univariate spline [61] fitting to the merged edges of each tow. Each tow edge is a list of pixels strictly ascending in terms of horizontal positions. The function (f) is estimated such that, given the horizontal

positions of the edge pixel (x), it can approximate the vertical position of the edge pixel ($y = f(x)$).

Defect Segmentation

The estimated boundaries of the tows can be processed to segment gaps and overlaps on the input image. When the upper boundary of the lower tow is higher than the lower boundary of the upper tow, it is possible that an overlap between the two tows happened. On the other hand, if the former is lower than the latter, a gap may exist between the tows corresponding to those curves. Based on these criteria the pixels in the input image can be classified into one of three classes neutral, gap, or overlap. Note that the segmented areas are identified as potential manufacturing defects. In fact, gaps and overlaps within the acceptable tolerance levels are not considered defects. Further investigations are necessary to determine whether these identified gaps and overlaps should be classified as defects or not.

Performance Evaluation

To assess the findings, the segmentation results are compared to the ground truth. The ground truth is established by an operator who labels the images. This labeling process involves using a brush tool to mark all the pixels that belong to the gap and overlap categories. To evaluate the performance of the gap/overlap segmentation algorithm, Intersection over Union (IoU) is utilized as a metric for each of these categories. After finding the IoU values for gaps and overlaps, the average of these two numbers is calculated which serves as an overall measure of the algorithm's performance.

3.3 Anomaly Detection

Figure 3.8 summarizes our anomaly detection framework. First, a composite scan is processed and local samples are extracted from the images. Then, the trained autoencoder generates an anomaly map, used to detect and locate the defects in the image.

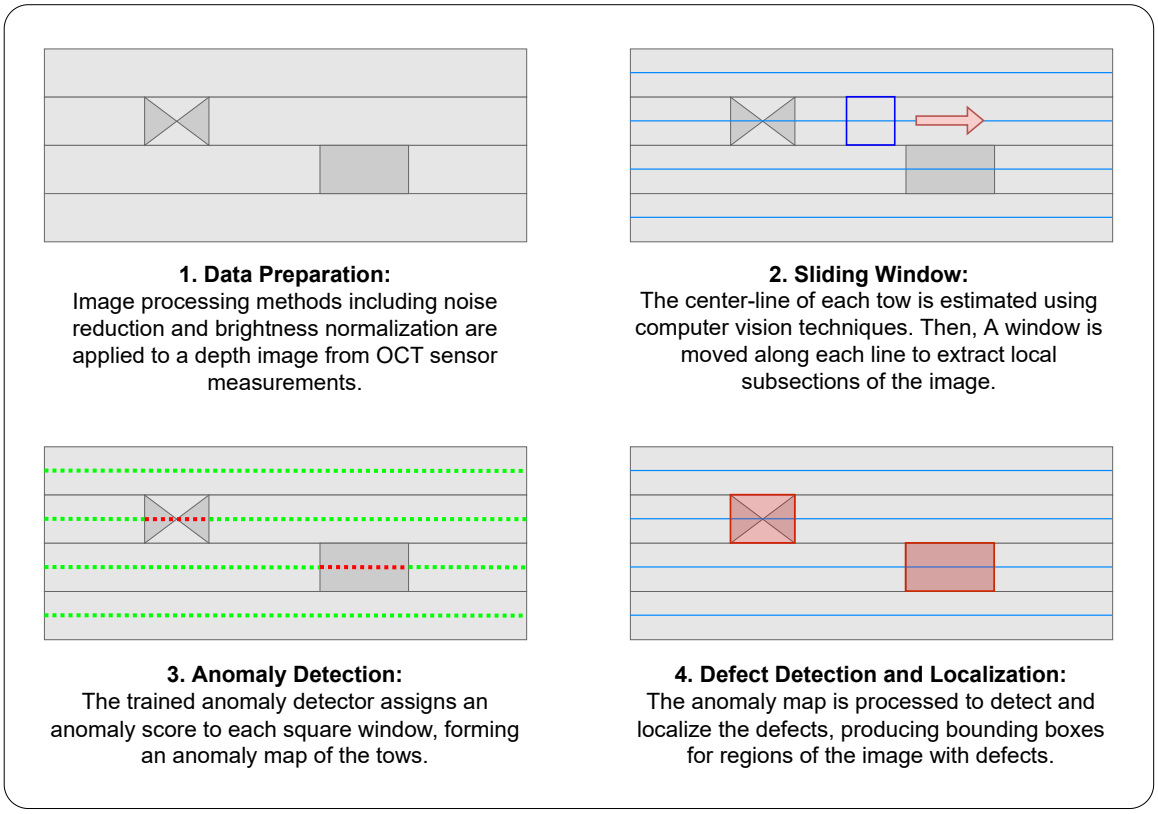


Figure 3.8: An overview of the proposed anomaly detection process shows the necessary steps.

Efficient Sliding Window

One of the most basic methods to detect local objects in an image is to move a window on the image and classify the smaller region inside the window [62]. This method is known as a sliding window in computer vision literature. In general, multiple scales of the sliding window are applied to the image to detect objects of different sizes. Also,

the window slides vertically and horizontally over the entire image to extract local samples. This generates many samples to be processed, increasing the computational costs. However, to analyze the tow-level defects in AFP, the scale of defects is limited to the width of the tows. Also, the sliding window only needs to move over each composite tow. These constraints allow us to use a more efficient sampling method, significantly reducing the computational costs and increasing the detection speed.

Each window detected as an anomaly is considered to contain a defect. In contrast, the windows that are not anomalies are assumed to have normal tow structures. Section 3.3 explains distinguishing between normal and abnormal samples.

The proposed sliding window moves along imaginary lines passing the center of composite tows (also known as centerlines). The centerlines create a skeleton that directs and constrains the sliding window's ROI. Section 3.3 describes the method to estimate the centerlines.

Tow Centerline Estimation

Hough transform is used as a line detector to estimate the vertical and horizontal limits of the tows. The conventional Hough transform searches for all the lines with stronger edges, but to detect the limit of the tows only horizontal and vertical lines are required. Original Hough transforms would find too many lines that do not represent the tows' boundaries. On the other hand, the depth maps generated from the composite surfaces have a specific structure. For example, all tows are in a straight horizontal line in the images, and the number of tows and their width is known. By incorporating this knowledge, the same Hough transform idea but with some modifications can detect the desired lines.

The Hough space is limited to a certain angle range, corresponding to the slope of the target lines. In Equation 3.5, h is the Hough space. ρ and θ denote the polar

distance and the angle, respectively. The angle ($theta$) is within ϵ tolerance of the goal angle θ_0.

$$h_{ij} = f(\rho_i, \theta_j) \quad \text{and} \quad \theta \in [\theta_0 - \epsilon, \theta_0 - \epsilon] \tag{3.5}$$

In a conventional Hough line detector, the local maxima in the 2D Hough space are obtained to find the lines' distance (ρ) and angle (θ). However, tow limit detection approach only requires horizontal ($\theta = 90°$) and vertical ($\theta = 0°$) lines. Therefore, the polling becomes a simpler problem of 1D maxima detection. To do so, the 2D Hough space is accumulated along the angle (θ) axis according to 3.6. This results in a 1D signal of distance (d).

$$d_i = \sum_j h_{ij} \quad \text{for} \quad \theta_j = 0° \text{ or } 90° \tag{3.6}$$

Then, a moving average filter with the size w is applied to the distance signal for noise reduction. Finally, the local maxima of the 1D signal are calculated(3.7). There are two additional constraints for finding the local maxima as shown in 3.8. First, the value of local maxima should be greater than a minimum value (d_{min}) to neglect weak lines. Second, the distance between two local maxima should exceed a minimum threshold (Δx_{min}) to prevent detecting overlapping lines multiple times.

$$x = argmax(d_i) \quad \text{and} \quad d_x = max(d_i) \tag{3.7}$$

$$d_x > d_{min} \quad \text{and} \quad |x_j - x_k| > \Delta x_{min} \tag{3.8}$$

The calculated local maxima (d_x) represent the distance of the detected lines from the origin, which is used to draw the detected horizontal and vertical lines.

Figure 3.9 demonstrates the process of using customized Hough transform to detect the horizontal and vertical lines. It shows that the algorithm can detect all the desired lines without any non-target lines being detected.

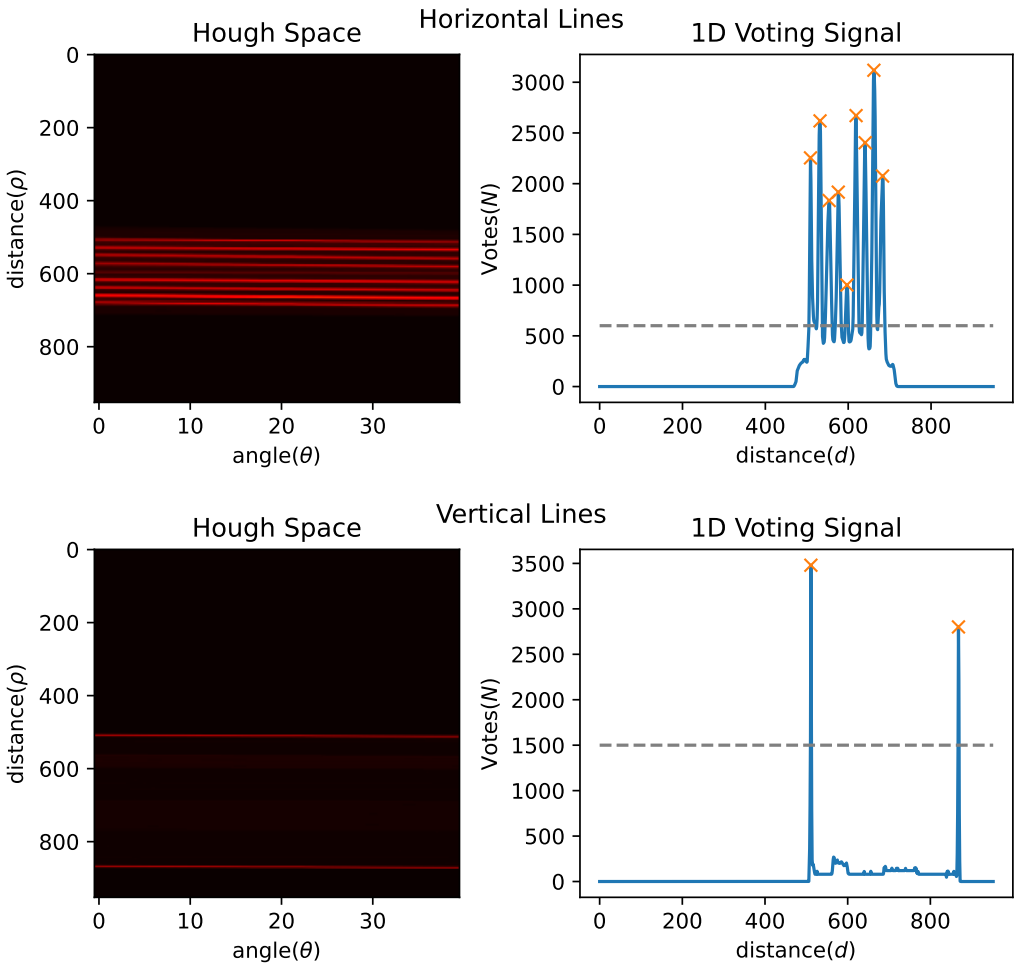
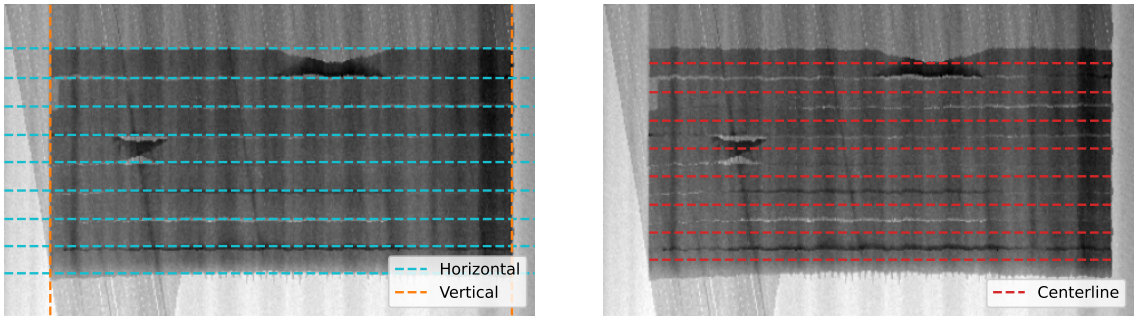


Figure 3.9: Customized Hough Transform for Detection of Horizontal and Vertical Lines. Left: Hough Transform Space with Limited Angles, Right: 1D Distance Signal

After detecting the boundaries of the tows, their center (centerlines) are calculated by averaging each two consecutive horizontal lines bounded within the detected vertical lines. This process is illustrated in Fig. 3.10.



(a) (b)

Figure 3.10: The centerline detection procedure contains two main steps: detecting horizontal and vertical lines (a) and estimating tow centerlines from the detected lines (b).

Supervised Learning

As defects are rare in AFP-manufactured composite parts, the defective samples are insufficient for training a model using supervised learning. An unbalanced dataset consisting of many non-defective samples and only a few defective samples increases the chance of the classifier being biased toward the non-defective class. Also, a balanced dataset with a few samples from both abnormal and normal classes under fits the model. To compare the results of the supervised vs. the proposed unsupervised learning approach, this work designs and tests the Convolutional Neural Network model presented in Table 3.1.

Unsupervised Learning

As mentioned in Chapter 1, autoencoders have shown great success at identifying anomalies in images. An autoencoder is an unsupervised learning model that reconstructs the given input by learning to minimize the error between the input and reconstructed output. They do this by encoding the input to a vector of latent features, also known as the bottleneck, and then decoding those latent features to

Table 3.1: Supervised Learning Network Structure

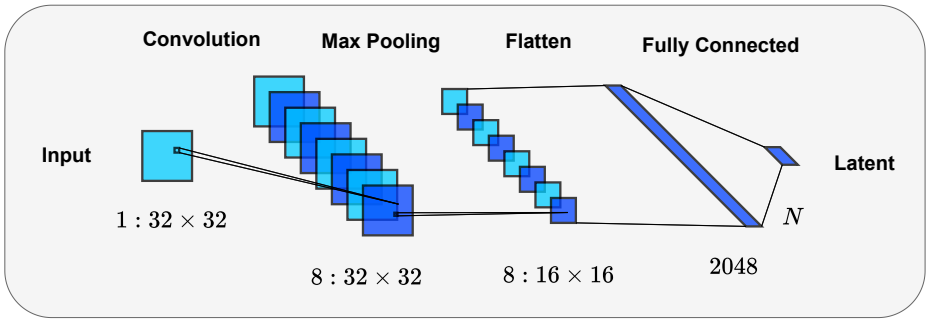
Convolutional Neural Network		
Layer	Layer Type & Activation	Output Shape
1	2D Convolution (ReLU)	(30, 30, 32)
2	2D Max Pooling	(15, 15, 32)
3	2D Convolution (ReLU)	(13, 13, 64)
4	2D Max Pooling	(6, 6, 64)
5	2D Convolution (ReLU)	(4, 4, 64)
6	Flatten	(1024)
7	Flatten	(64)
8	Fully Connected	(2)
9	Soft Max	(2)

reconstruct the input. Convolutional Autoencoders (CAEs) are a group of autoencoders that use convolution layers in their network structure. Convolutional Neural Networks (CNNs) are more popular for image-based autoencoders than basic fully-connected networks. This is because CNNs incorporate receptive fields using kernels that maintain the spatial relationships of the data. CNNs are also computationally efficient with sparse connectivity of neurons.

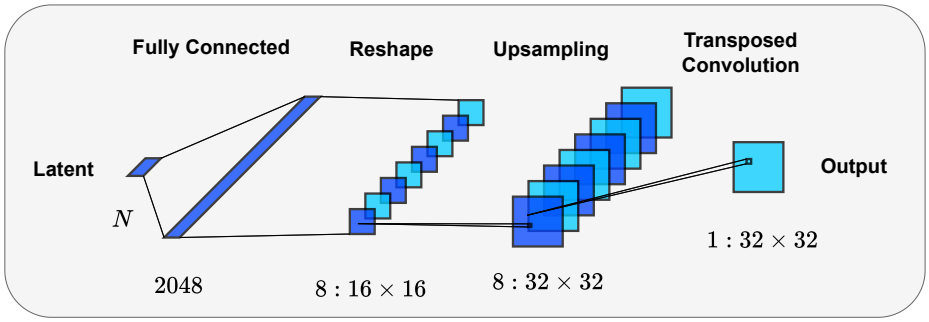
If only normal samples are used to train the autoencoder, it will be able to reconstruct similar normal samples accurately, and the reconstruction results for abnormal samples will be poor. Therefore, reconstruction error can be used as an indicator of how anomalous each input is. For inference, each cropped window of a composite material depth map is fed into the trained autoencoder. The reconstruction error of each window is then used as an anomaly score to create an anomaly map for the entire image. Reconstruction error of a window centered at (x,y) is calculated using Eq. 3.9 in which $p_{i,j}$ and $\hat{p_{i,j}}$ are the pixel value of the input and reconstructed output, respectively, and b is half of the size of each window.

$$m_{x,y} = \frac{1}{(2b)^2} \sum_{i=x-b}^{x+b-1} \sum_{j=y-b}^{y+b-1} [p_{i,j} - \hat{p}_{i,j}]^2 \tag{3.9}$$

In this work, a CAE is designed and used as the anomaly detector. The design incorporates symmetric encoder and decoder structures shown in figures 3.11a and 3.11b respectively. For training the model, Mean Squared Error (MSE) is employed as the loss function. Although the full method uses continuous valued anomaly maps to identify the defects rather than a binary prediction, binary classification can still be useful in validating the model performance. For this, a threshold parameter is introduced to classify samples based on their reconstruction error. To select the threshold value, a Receiver Operating Characteristic (ROC) curve is applied. The ROC curve plots the true positive rate against the false positive rate while varying the threshold value. In an ideal case, the selected threshold would give a true positive rate of 1 and a false positive rate of 0. In the ROC plot, this corresponds to the upper left corner and hence the best threshold value is selected from the curve at the point closest to that corner.



(a) Encoder



(b) Decoder

Figure 3.11: A graphic depicting the network structure of the proposed autoencoder. Above is the encoder structure (a), and below is the decoder structure (b).

Defect Localization

The anomaly detection generates an array of anomaly scores for each tow, which can be considered as a 1D digital signal. Any area of this signal with a concentration of high values indicates the presence of a defect. In Computer Vision literature, these areas are called blobs [63, 64]. Difference of Gaussians (DoG) algorithm [65] is used to detect the blobs. In this approach, the signal ($f(x)$) is filtered using Gaussian kernels with increasing values for standard deviations (σ) as described in Equation 3.10, in which $*$ denotes a convolution operator. Then, the subtractions of each two successively filtered signals are calculated. The local maxima of $g(x, \sigma)$ represents the blobs. In such maxima points, x and σ correspond to the location and characteristic scale (size) of the blob, respectively.

$$g(\sigma, x) = \sigma^2 \frac{\partial n_\sigma^2}{\partial x^2} * f(x) \tag{3.10}$$

For each defect, two parameters are detected, radius and center. With this information, the detected blobs can be transferred from the anomaly map to image space.

Chapter 4

Experimental Setup

In this chapter, detailed insights are provided regarding the preparation of the dataset and the procedures involved in training and testing the machine learning models. Additionally, essential implementation details and design parameters are included, enabling the replication of the thesis results.

4.1 Dataset and Training

While none of the approaches presented in this thesis necessitate defect samples for their functioning, the evaluation of their performance requires images containing manufacturing defects. However, due to the infrequent occurrence of defects in AFP, there is a scarcity of defect samples even for evaluation purposes. To overcome this limitation, defects are manually introduced into the composite parts. An expert peels off the composite tapes after each layer is attached, and manufacturing defects are intentionally added to these tapes through various operations, such as twisting them.

The dataset used in the anomaly detection approach consists of 44 depth maps of composite surfaces, among which only two include twists and fold manufacturing defects. Using the presented efficient sliding window method the local samples

Dataset	Normal Samples	Abnormal Samples
Training	27406	0
Test	1648	42

Table 4.1: Number of Samples for Training the CAE

(patches) are extracted from these images. Some of these samples are demonstrated in Figure 4.1. Table 4.1 shows the number of extracted samples. As shown in this table, most extracted patches belong to the normal class which are split for training and test propose. The few defective samples are only used to test the anomaly detector's performance.

The dataset utilized in the anomaly detection approach comprises 44 depth maps of composite surfaces, of which only two include twists and fold manufacturing defects. Through the efficient sliding window method presented, local samples (patches) are extracted from these images. Figure 4.1 illustrates some of these samples. Table 4.1 displays the number of extracted samples which are used to train the Convolutional Autoencoders. As shown in the table, the majority of the extracted patches belong to the normal class, which is divided into training and testing subsets. The limited number of defective samples is solely used to evaluate the performance of the anomaly detector.

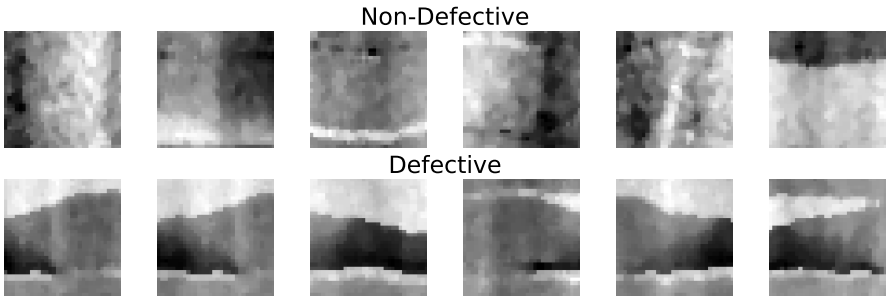


Figure 4.1: A dataset is created from cropped sections of the depth maps using the sliding window method.

4.2 Design Parameters

To eliminate the salt and pepper noise dots, a median filter with a kernel size of $N \times N$ is employed. For optimal results, a moderate filter size of ($N = 3$) is used to effectively remove the majority of these dots while preserving the essential visual details within the image.

Based on the detected centerlines, a square window slides across the tows to extract cropped regions. We select a window size of 32×32 pixels to cover approximately 1.5 times the width of composite tapes. In this implementation, a stride of 8 pixels is used to move the window and sample the information cropped inside. This combination of window size and stride allows enough overlap between the nearby samples while keeping the samples sufficiently distinctive. Some extracted samples are presented in Fig. 4.1.

The autoencoders are trained with a dataset consisting of 27406 only normal samples. An Adam optimizer is employed with an MSE loss function to train the network. The batch size is set to 128. Each autoencoder undergoes training for 50 epochs, completing in under 5 minutes on a computer with the following specifications:

- **Processor (CPU):** Intel(R) Xeon(R) E5-1607 v4 @ 3.10GHz
- **Graphics (GPU):** NVIDIA GeForce GTX 1080
- **Memory (RAM):** 32.0 GB

Chapter 5

Results and Discussion

In this chapter, a detailed presentation of the results is provided for both proposed approaches in defect detection for AFP. The findings are presented in a step-by-step manner, showcasing the outcomes and achievements of the implemented methodologies.

5.1 Image Preprocessing

In Figure 5.1 the impact of median filters on the image is demonstrated. Using a large median filter ($N = 6$) results in losing important features like the edges of the tows. Whereas, choosing an average median filter ($N = 3$) reduces salt and pepper noise while preserving the edges of the tows.

5.2 Gap and Overlap Segmentation

Figure 5.2 shows how a canny edge detector extracts the edges on the image. By increasing the value standard deviation (σ) for the Gaussian filter, more clear representation of tow edges is extracted but some edges are missed.

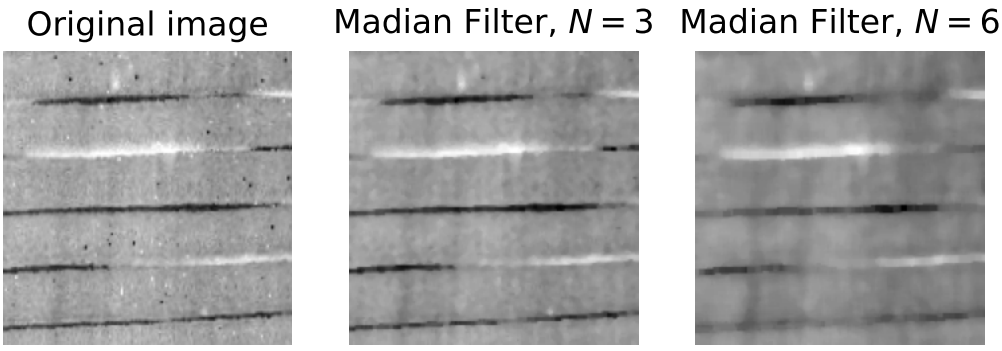


Figure 5.1: Applying median filters with different window sizes

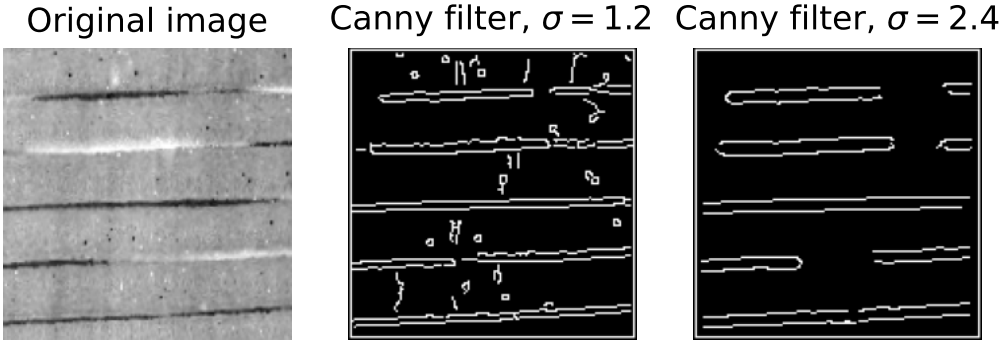


Figure 5.2: Canny edge detection with different Gaussian filter sizes

To filter out unwanted edges, a morphological opening operator is used that only preserves the horizontal edges. The detected edges before and after applying the morphological operation are demonstrated in Figure 5.3.

Figure 5.4 illustrates the result of tow edge classification. The Sobel operator generates the horizontal gradient of the image. Green and red colors represent positive and negative gradient values. Also, The pixel brightness on the gradient image shows the normalized magnitude of the gradient. The gradient values are positive for the lower edges of the tows and negative for the upper edges of the tows. Considering the sign of the horizontal gradient, the edges are classified into either upper or lower edges of the tows which are distinguished by green and red colors, respectively.

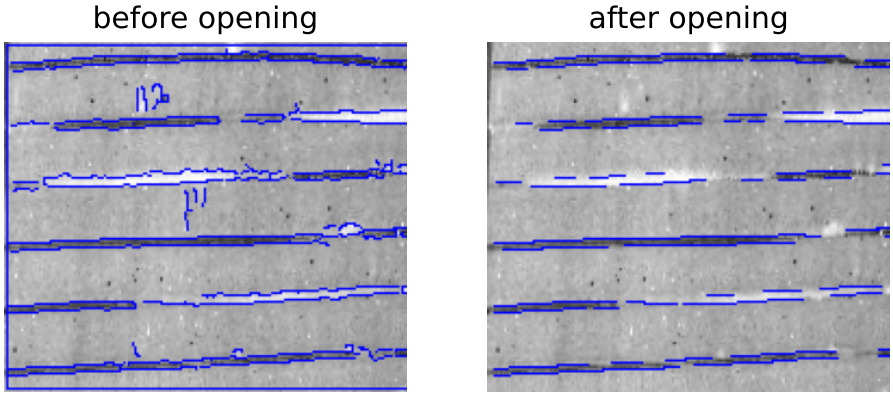


Figure 5.3: Edge detection before and after opening operation

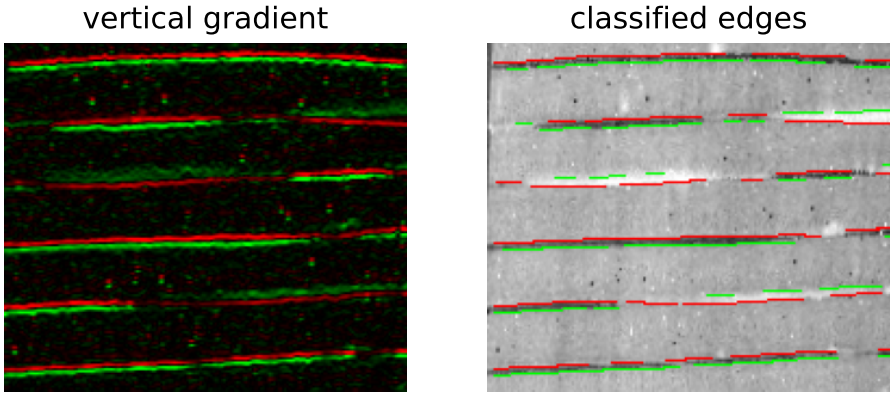


Figure 5.4: Positive and negative components of the horizontal gradient (left) are used to classify the edges into top and bottom edges of the tows (right).

Figure 5.5 demonstrates the result of edge grouping for the upper tow boundaries. The partitioning algorithm uses a customized distance to group the edge pieces into equivalency classes. Each edge is annotated with the detected equivalency class number. As expected, the edge pieces in the same tow are grouped into the same class.

Figure 5.6 illustrates the result of horizontal edge reconstruction. The upper and lower edges of the tows are interpolated by a cubic spline and annotated on the original image. The areas between these edges are also annotated as gaps or overlaps.

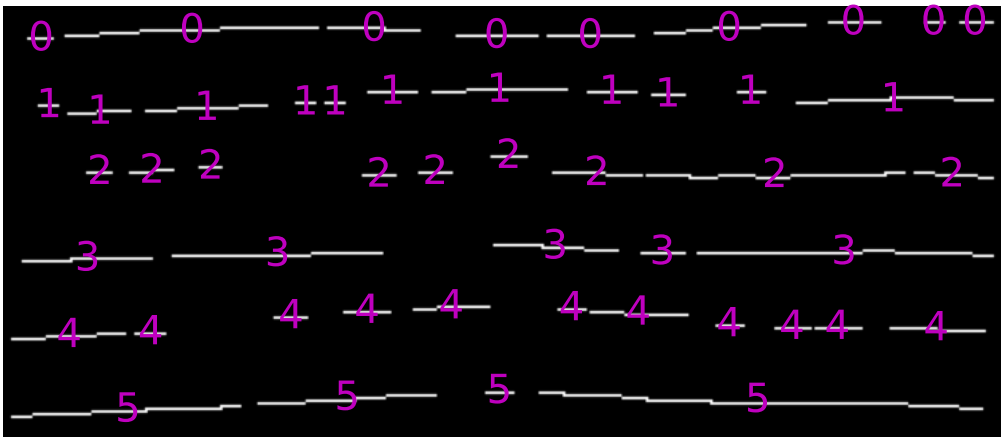


Figure 5.5: Grouping Edge Regions. Each region is labeled with the corresponding group number.

The software was unable to detect part of the edges on the left and right sides of the tows. This usually happens when the two consecutive tows are perfectly laid close to each other without any gaps or overlaps. In that case, no gaps or overlaps are missed.

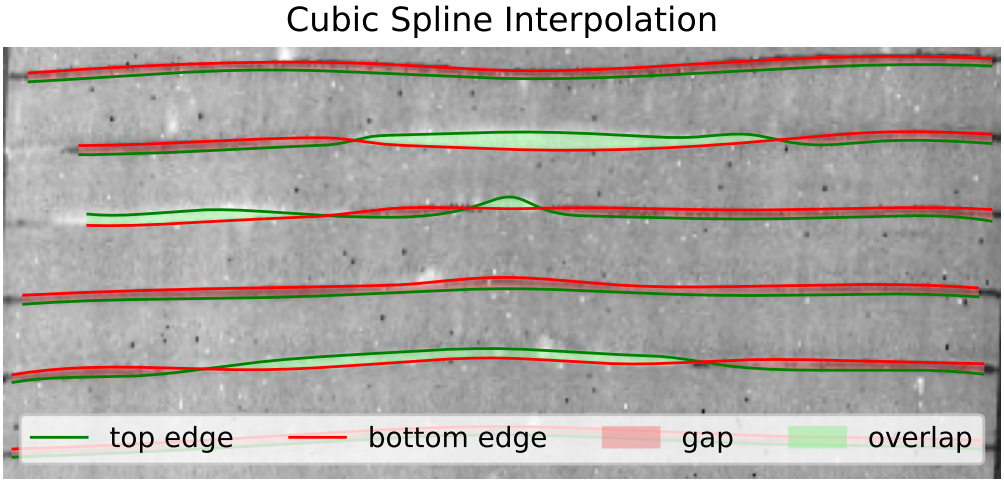


Figure 5.6: Reconstructed horizontal boundaries of tows reveal gaps and overlaps.

A comparison between the segmented image and annotated ground truth is demonstrated in Figure 5.7. The values of Intersection over Union (IoU) are 0.381, and 0.494

for the gap and overlap classes, respectively. This is equal to an average IoU of 0.438 for both classes.

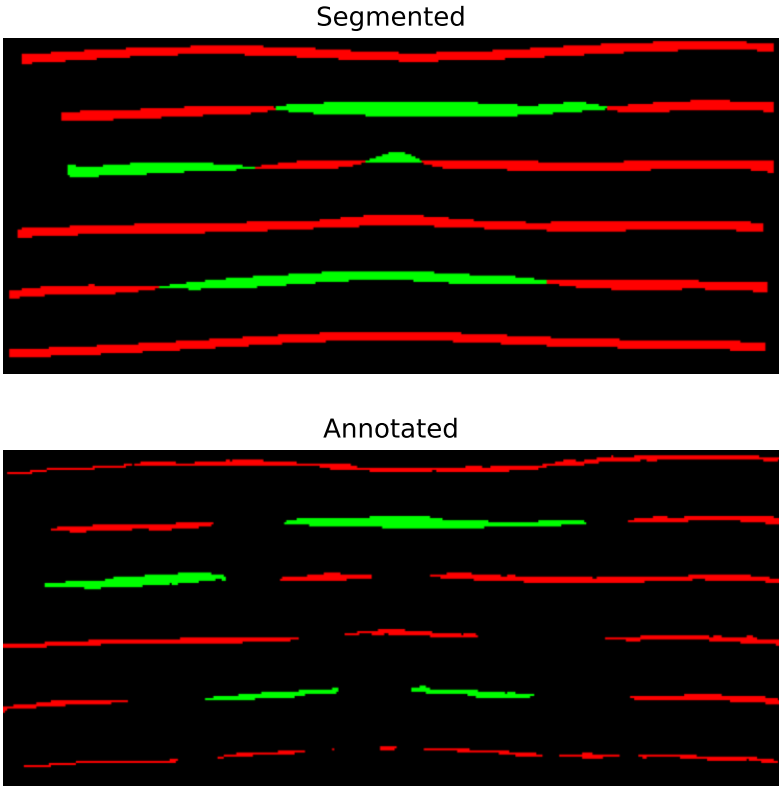


Figure 5.7: A comparison between gap and overlap: predicted segmentation (top) and annotated ground truth (bottom). Gaps are shown in red and overlaps in green.

5.3 Anomaly Detection

The performance of the anomaly detection and localization system depends on two factors. First is the number of samples the anomaly detector is correctly classifying as normal or abnormal. Second is the size and location accuracy of the predicted defects. This section evaluates these two aspects.

First, the performance of the supervised learning method was evaluated. The confusion matrix of the test set is shown in 5.8. The model has a high classification

accuracy on the training set but a low one on the test set. This means that the model is over-fitted and instead of learning the spatial pattern in the input images, it memorized the samples in the training set. Consequently, the model is confused while classifying an unseen sample.

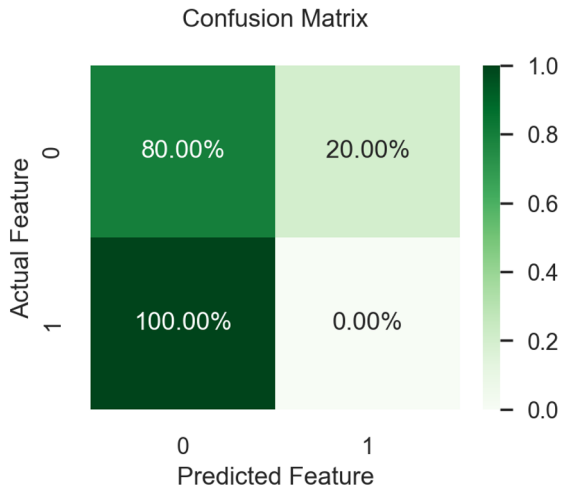


Figure 5.8: Confusion Matrix in Supervised Learning Method

The network structure proposed in Fig. 3.11 is implemented using three different latent dimensions 2, 16, and 128 for comparison. Each network is trained with a dataset consisting of only normal samples. The curves in Fig. 5.9 demonstrate the training losses of each autoencoder trained over 50 epochs. Comparing the training loss curves shows that the models' reconstruction ability improves with higher dimensional latent space. The curves also show that the models learn relatively quickly with little improvement in the later epochs.

Reconstruction results for the autoencoders are demonstrated in Fig. 5.10. The original samples are randomly selected from normal and abnormal classes in the test set. These results clearly show the improved reconstruction performance with a higher dimensional latent space. It shows that the autoencoder with a 128-dimensional latent vector is able to produce good reconstructions for both normal and abnormal samples.

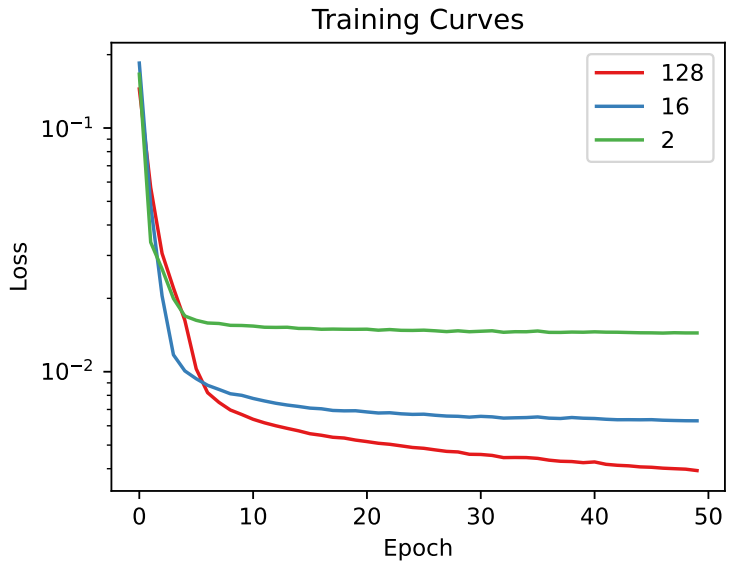


Figure 5.9: The training MSEs of the three autoencoders are plotted in comparison over 50 epochs.

The 16-dimensional autoencoder, on the other hand, produces relatively good reconstructions for normal samples and poorer reconstructions of defect samples. This is ideal for the classification method to distinguish anomalies. Finally, the autoencoder with only a 2-dimensional latent space is unable to make good reconstructions for any of the input samples.

Fig. 5.11 shows comparisons of the reconstruction error. In figure 5.11a the distributions of Mean Squared Error are shown for the 16-dimensional autoencoder on the training set and test set. Note that the training set only includes normal samples whereas the test set contains both normal and abnormal samples, separated accordingly. As the figure suggests, the normal samples have a similar distribution in both the training and test sets. On the other hand, the abnormal samples have a generally higher MSE with a slight overlap on normal sample distribution. In an ideal case, if there were no overlap between these two distributions, a threshold would exist that serves as a decision boundary to classify the samples into normal and abnormal categories. With existing overlap, however, an ROC curve can help to select the decision

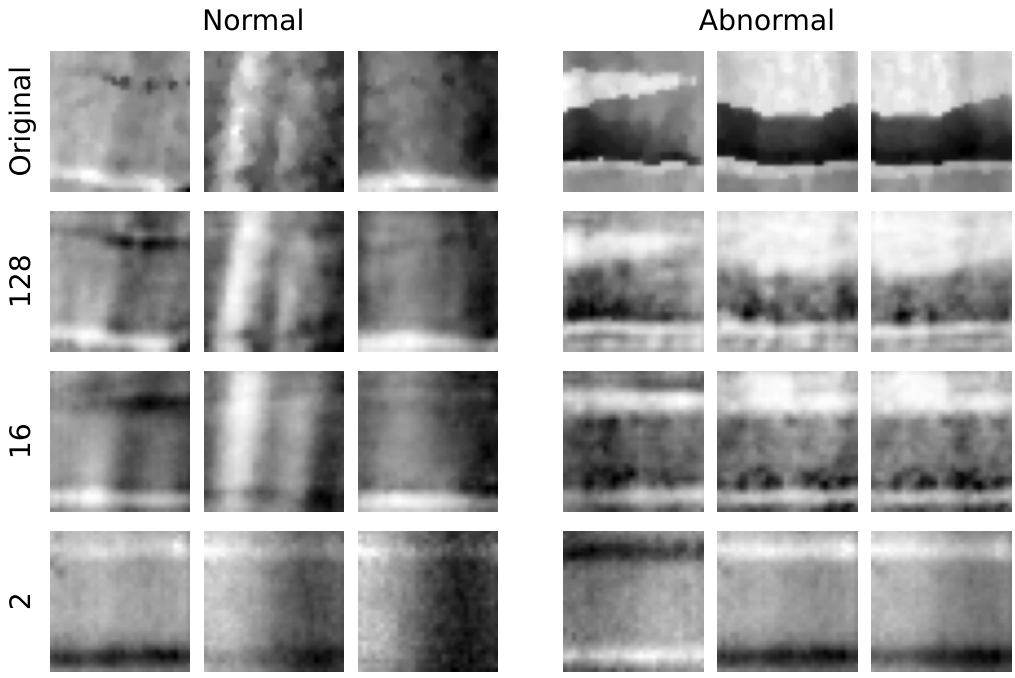
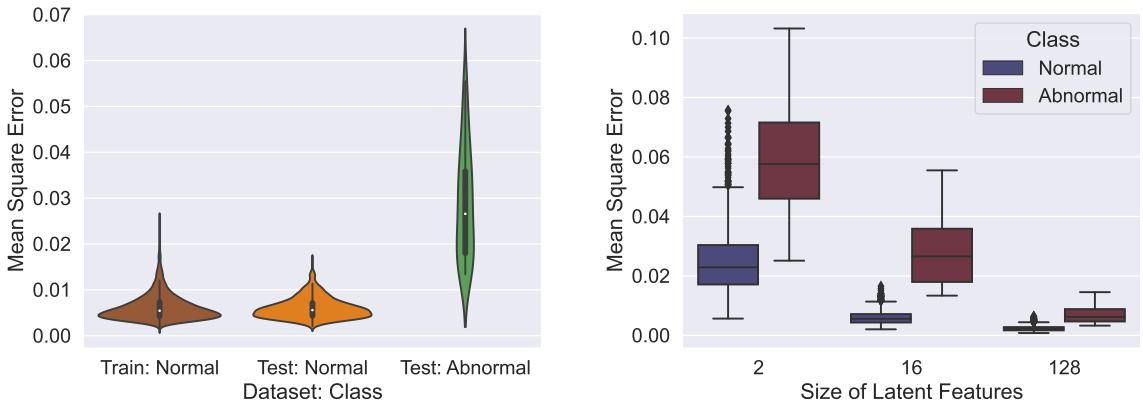


Figure 5.10: The resulting reconstructions from the autoencoders with various latent sizes are compared for both normal and abnormal test samples.

boundary that makes the best trade-off between true and false positive rates. Taking a closer look at the difference between the three autoencoders, Fig. 5.11b shows boxplots of the MSE for normal and abnormal samples in the test set, using different number of latent features. Here it shows how separable the two classes are based on reconstruction error alone. For the 2-dimensional autoencoder, the interquartile ranges are separable, but there is a significant overlap when considering the whiskers. For the 16-dimensional autoencoder, the separation is greatly improved with minimal overlap between the whiskers. The 128-dimensional autoencoder, however, does not show significant separation and would be impossible to accurately classify the two classes based on MSE alone.

Fig. 5.12 shows the ROC curves for each of the three autoencoders. The best threshold should lead to the lowest false positive rate and highest true positive rate. Therefore, the points on the curves that are closer to the point $(0, 1)$ are the best



(a) Comparing Test and Training

(b) Comparing Different Latent Sizes

Figure 5.11: Distributions of MSE for different input types and latent sizes are presented for comparison.

Table 5.1: Test Metrics of Convolutional Autoencoder

Latent Dim.	AUC	Best Threshold	Precision	Recall	F1-Score
128	0.978	3.86	0.918	0.929	0.923
16	0.999	13.69	0.992	0.976	0.984
2	0.942	42.08	0.922	0.810	0.862

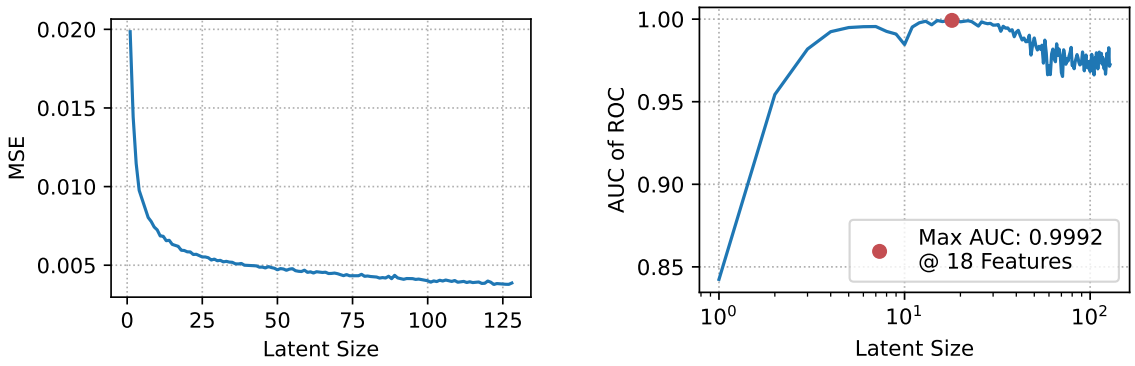
thresholds, which are denoted using stars. These results further demonstrate that classification performance does not correspond with reconstruction performance, as the 2D and 128D autoencoders' ROC curves have worse classification performance than the 16D model. The best-performing model is the autoencoder with a 16-dimensional latent space, achieving a high true positive rate with a low false positive rate.

Classification results of each autoencoder with the selected threshold values found by the ROC curves are summarised in Table 5.1. The table reports precision, recall, F1 score, area under the ROC curve (AUC), and selected threshold for each classifier.

Fig. 5.13 shows the effect of the latent vector size on the performance of the model. Fig. 5.13a suggests that larger latent dimensions will produce lower recon-

Figure 5.12: ROC curves of the test set are plotted for the three autoencoder classifiers.

struction errors. However, an accurate classification model does not require the lowest reconstruction error, but a moderate reconstruction that leads to better classification performance. In Fig. 5.13b the best latent dim is found by calculating the minimum AUC of the ROC curve while varying the latent dim.



(a) MSE

(b) AUC of ROC

Figure 5.13: MSE and AUC of ROC are plotted for autoencoder models with latent dimensions varied from 1 to 128.

Fig. 5.14 shows the classification confusion matrix while using the optimal latent

size. The off-diagonal values in this matrix are low which shows that most of the samples from both normal and abnormal classes are classified correctly.

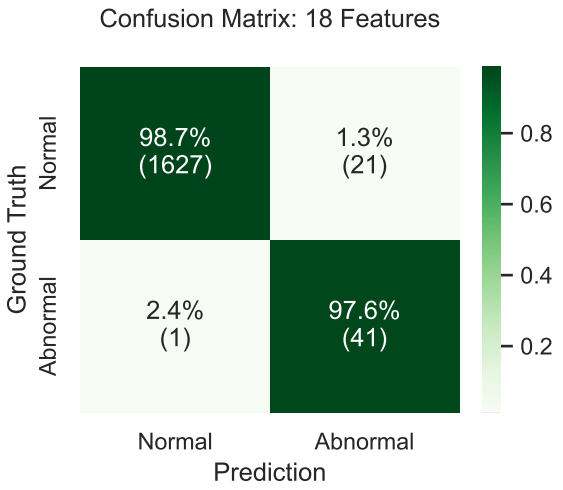


Figure 5.14: The confusion matrix demonstrates the evaluation of the best model.

Fig. 5.15 illustrates the results of the anomaly detector on a 2D depth map. The color of each point indicates the normalized MSE for reconstructing a small window around the point with the anomaly detector. As can be seen, the defective areas have a large density of points with a higher MSE. This information can be used to detect these areas while ignoring individual outlier values.

In Fig. 5.16 the process of detecting the defects from the anomaly map is illustrated. The elevation in each curve represents the MSE value for one tow (represented by color in the previous figure). The arrows show the detected blobs after applying the Derivative of Gaussians method. It is observed that only the areas with an extended length of high MSE values are detected as blobs.

Fig. 5.17 shows the final output of the anomaly detection approach, comparing the annotated defect bounding boxes with the predicted bounding boxes. To evaluate this result, a popular metric in object detection, known as mean Average Precision (mAP), is used. This metric requires assigning a value of classification confidence

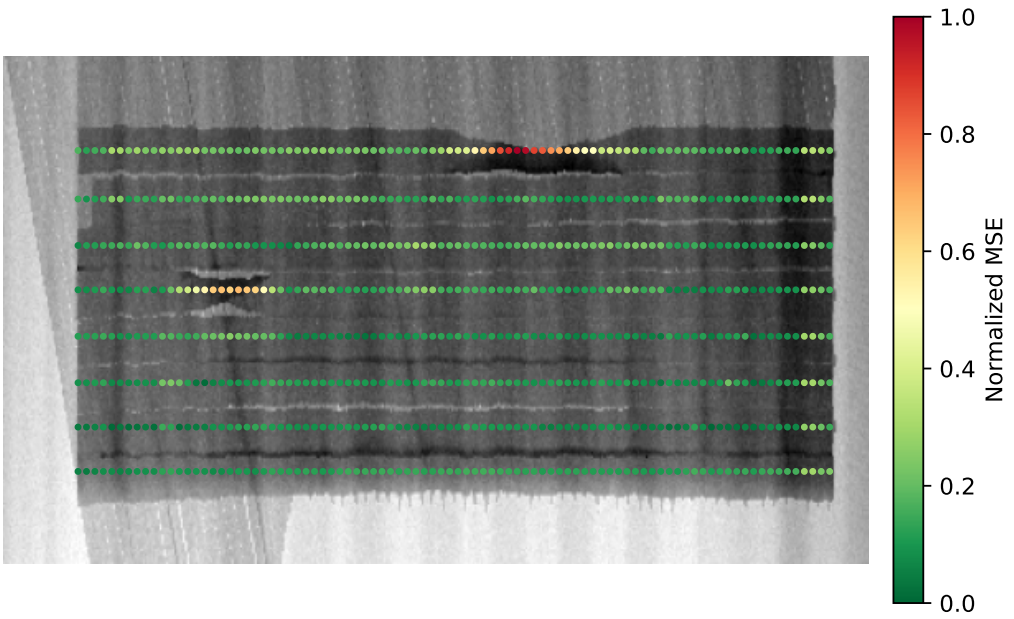


Figure 5.15: An anomaly map is generated from the MSE of individual cropped windows.

to the detection. In this study, the confidence score of detection is defined as the summation of all the MSE values within the region of the corresponding detected blob. Also, a minimum value for Intersection over Union is needed to consider a prediction valid. Because, in the context of AFP inspection, the precise location of the defects is not as important as identifying them, a lenient IoU of 0.5 is employed as the minimum threshold for a prediction to match a ground truth bounding box. The network achieves an mAP of 0.85 on the test images. This is sufficient for the task at hand, especially given that this method only requires a small set of normal scans.

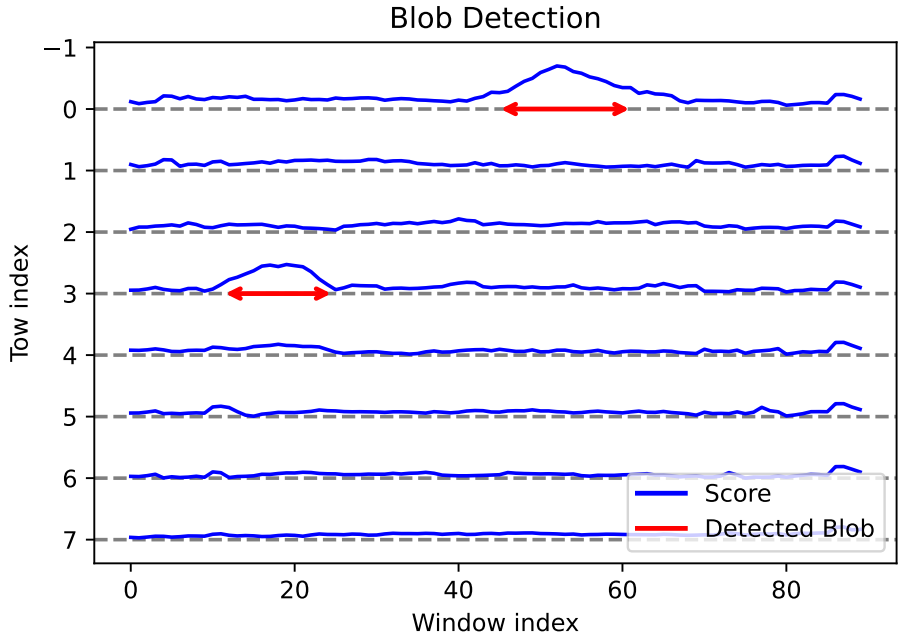


Figure 5.16: Anomaly scores are visualized as 1D signals for blob detection.

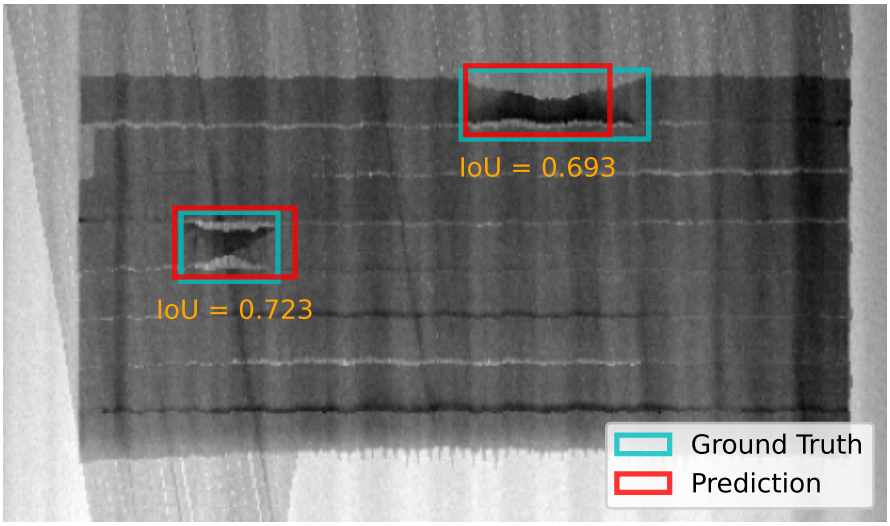


Figure 5.17: Predicted bounding boxes are displayed on the original depth map compared to the ground truth bounding boxes. Also, the values of Intersection over Union are displayed.

Chapter 6

Conclusions and Future Directions

This thesis tackles the challenges associated with training data for defect detection tools and presents alternative approaches to address them effectively.

Defect detection plays a vital role in the manufacturing industry. Many automated defect detection systems rely on end-to-end supervised learning methods, which necessitate a substantial dataset of labeled defect samples. However, manufacturing data is often scarce, and obtaining a sufficient number of defective samples is challenging. Additionally, the annotation of manufacturing defects is a time-consuming process that is prone to human errors. These difficulties make end-to-end supervised learning methods inefficient. To overcome the limitations of conventional supervised learning, this study proposes two practical solutions:

1. **Unsupervised Anomaly Detection:** This approach eliminates the need for labeled defect samples and instead relies on a smaller subset of normal samples.

2. **Rule-based Computer Vision:** This method eliminates the need for training data or labeling altogether.

By implementing these alternative approaches, the objective is to effectively tackle the limitations related to training data for defect detection tools. To demonstrate the

feasibility of these solutions, this work specifically concentrates on the application of Automated Fiber Placement (AFP). However, it is important to note that similar approaches can be employed in other visual inspection applications as well. This highlights the versatility and applicability of the proposed solutions beyond the scope of AFP.

Proposed Defect Detection Approaches in AFP

Automated Fiber Placement involves the manufacturing of composite fiber parts using robotics and intelligent systems. This process is susceptible to various types of defects such as twists, gaps, overlaps, and more. These defects can significantly impact the final product's quality, necessitating thorough inspection of the composite parts. Manual human inspection has traditionally been employed for this purpose, but it is time-consuming, labor-intensive, and prone to human errors. To enhance the efficiency, accuracy, and reliability of the AFP process, the development of an automated inspection system capable of defect detection is crucial. Over the past few decades, numerous inspection tools have been designed for AFP and similar industries. Many of these tools utilize profilometry technologies like laser scanning, thermal imaging, and optical sensors to generate visual measurements of the part's surface. The data used in this thesis is obtained from a laser scanner that operates based on Optical Coherence Tomography (OCT) technology.

The proposed solutions accept 3D point cloud scans of the composite part as input. These scans are then converted into 2D depth map images and preprocessed to prepare them for the defect detection process. The defect detection system generates detection results, which are incorporated into the depth map images as annotations. This enables technicians to easily locate, investigate, and rectify the identified defects.

The first solution introduces a practical and innovative method for anomaly de-

tection in AFP, offering a way to overcome the challenge of data scarcity by focusing solely on normal samples and identifying any deviations as potential defect candidates. This computer vision approach employs the Hough Transform to detect individual composite tapes (tows) and generates a dataset of sub-images by sliding a window along the center of each tow. The extracted data is then utilized to train an autoencoder specifically designed for anomaly detection. In this proposed method, the autoencoder is optimized to have an optimal number of latent features that enable the best differentiation between normal and abnormal samples. To assess its effectiveness, the autoencoder is evaluated in a classification task and achieves a per-class accuracy of over 97% for both normal and defective samples. Subsequently, the sub-images are processed to generate an anomaly map of the entire image. To localize the anomalous regions, this map undergoes further processing using a blob detection algorithm, resulting in the creation of bounding boxes around the detected anomalies.

The second approach introduces a computer vision algorithm designed to detect and segment gaps and overlaps, which are the most prevalent defects in Automated Fiber Placement (AFP). This is done by utilizing classical image processing and computer vision techniques. The first step is the extraction of the image edges, which contain essential information about the tows. The algorithm employs morphological operations to filter the horizontal edges and uses the image gradient to classify them as either the upper or lower edges of the tows. These edges are then grouped and merged based on the tow to which they belong. The merged edges are utilized to generate a spline representation of the tow boundaries. Finally, analyzing the boundaries of consecutive tows leads to the identification of any potential gaps and overlaps between them and the segmentation of the pixels associated with these defects. Furthermore, this thesis conducts an investigation into the impact of various design parameters on the segmentation tool's performance. To evaluate its effectiveness, the Intersection

over Union (IoU) metric is employed, resulting in an average value of 0.438 for the gap and overlap classes.

Future Work

This thesis showcases the practicality of the proposed defect detection approaches in AFP. The presented solutions address the need for a large labeled dataset, specifically including samples of defects. However, there are opportunities for further enhancing the performance of the inspection system and extending its applicability to broader domains. Here are some suggestions for future researchers to consider:

1. Given the constraint of limited real data, future work can delve into the exploration of data engineering techniques, including data augmentation and synthetic data generation. These approaches have the potential to enhance the quality and expand the quantity of the dataset. This can provide valuable resources for training and further improving the performance of defect detection systems in scenarios where obtaining large quantities of real data is challenging or impractical.

2. While the anomaly detector successfully identifies potential defect candidates, it does not classify the specific types of defects. To address this limitation, the generated bounding boxes around the detected anomalous areas can be used to extract training data for developing neural networks or other classification models capable of accurately categorizing various types of defects. By incorporating defect classification into the system, a more comprehensive and detailed analysis of the detected anomalies can be achieved.

3. Building upon the extracted gaps and overlaps, it is possible to develop an automated system that classifies them as either defects or non-defects. This

classification can be based on analyzing the shape and size characteristics of the identified gaps and overlaps. By incorporating such a system, a more comprehensive understanding of the severity and nature of these gaps and overlaps can be attained, allowing for more effective quality control and decision-making processes.

4. The parts produced in AFP exhibit a tape-by-tape structure. With suitable adaptations, the approaches utilized in this study can be extended to quality inspection tasks in other industries that possess a similar structural pattern. Examples of such industries include welding or gluing, where the inspection of tape-like structures is relevant. By leveraging the knowledge and methodologies developed in this research, advancements can be made in defect detection and quality assurance across a broader range of industries that share this common structural characteristic.

Chapter 7

Additional Information

7.1 Preface

Some parts of this thesis on the anomaly detection approach have been published as a pre-print on ArXiv, titled "Anomaly Detection in Automated Fibre Placement: Learning with Data Limitations" [66]. In this publication, I led the research and contributed to idea formation, programming, algorithm development, visualizations, and paper writing. Todd Charter, a graduate researcher at UVic, made significant contributions to the literature review, dataset pre-processing and organization, the autoencoder design and implementation, and manuscript editing. As the research lead at LlamaZOO Interactive Inc., Li Ji played an active role throughout the project, providing industrial insights, contributing to concept formation, and participating in paper revision. Maxime Rivard from the National Research Council of Canada (NRC) and Gil Lund from Fives Lund LCC collaborated on sensor data generation, providing valuable industrial background and proofreading assistance. Homayoun Najjaran, the supervisor, offered general feedback and supervision during the paper revision process.

Bibliography

[1] B. Böckl, A. Wedel, A. Misik, and K. Drechsler, "Effects of defects in automated fiber placement laminates and its correlation to automated optical inspection results," *Journal of Reinforced Plastics and Composites*, vol. 42, pp. 3–16, Jan. 2023. Publisher: SAGE Publications Ltd STM.

[2] A. Brasington, C. Sacco, J. Halbritter, R. Wehbe, and R. Harik, "Automated fiber placement: A review of history, current technologies, and future paths forward," *Composites Part C: Open Access*, vol. 6, p. 100182, Oct. 2021.

[3] L. Zhang, X. Wang, J. Pei, and Y. Zhou, "Review of automated fibre placement and its prospects for advanced composites," *Journal of Materials Science*, vol. 55, pp. 7121–7155, June 2020.

[4] M. Palardy-Sim, M. Rivard, S. Roy, G. Lamouche, C. Padioleau, A. Beauchesne, L.-G. Dicaire, D. Lévesque, J. Boisvert, M.-A. Octeau, and others, "Next Generation Inspection Solution for Automated Fibre Placement," in *The Fourth International Symposium on Automated Composites Manufacturing*, p. 64, 2019.

[5] W. Woigk, S. R. Hallett, M. I. Jones, M. Kuhtz, A. Hornig, and M. Gude, "Experimental investigation of the effect of defects in Automated Fibre Placement produced composite laminates," *Composite Structures*, vol. 201, pp. 1004–1017, Oct. 2018.

[6] S. Meister, M. A. M. Wermes, J. Stüve, and R. M. Groves, "Review of image segmentation techniques for layup defect detection in the Automated Fiber Placement process," *Journal of Intelligent Manufacturing*, vol. 32, pp. 2099–2119, Dec. 2021.

[7] F. Heinecke and C. Willberg, "Manufacturing-Induced Imperfections in Composite Parts Manufactured via Automated Fiber Placement," *Journal of Composites Science*, vol. 3, p. 56, June 2019. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[8] S. Hajizadeh, A. Núñez, and D. M. J. Tax, "Semi-supervised Rail Defect Detection from Imbalanced Image Data," *IFAC-PapersOnLine*, vol. 49, pp. 78–83, Jan. 2016.

[9] J. Jing, Z. Wang, M. Rätsch, and H. Zhang, "Mobile-Unet: An efficient convolutional neural network for fabric defect detection," *Textile Research Journal*, vol. 92, pp. 30–42, Jan. 2022. Publisher: SAGE Publications Ltd STM.

[10] H. He and E. A. Garcia, "Learning from Imbalanced Data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, pp. 1263–1284, Sept. 2009. Conference Name: IEEE Transactions on Knowledge and Data Engineering.

[11] X. Deng, Y. Ma, and M. Dong, "A new adaptive filtering method for removing salt and pepper noise based on multilayered PCNN," *Pattern Recognition Letters*, vol. 79, pp. 8–17, Aug. 2016.

[12] U. Erkan, L. Gökrem, and S. Enginoğlu, "Different applied median filter in salt and pepper noise," *Computers & Electrical Engineering*, vol. 70, pp. 789–798, Aug. 2018.

[13] K.-y. Tien, H. Samani, and J. H. Lui, "A survey on image processing in noisy environment by fuzzy logic, image fusion, neural network, and non-local means," in *2017 International Automatic Control Conference (CACS)*, pp. 1–6, Nov. 2017.

[14] B. I. Justusson, "Median Filtering: Statistical Properties," in *Two-Dimensional Digital Signal Prcessing II: Transforms and Median Filters*, Topics in Applied Physics, pp. 161–196, Berlin, Heidelberg: Springer, 1981.

[15] G. Gupta, "Algorithm for Image Processing Using Improved Median Filter and Comparison of Mean, Median and Improved Median Filter," *International Journal of Soft Computing and Engineering (IJSCE)*, 2011.

[16] A. Raid, W. Khedr, M. El-dosuky, and M. Aoud, "Image Restoration Based on Morphological Operations," *International Journal of Computer Science, Engineering and Information Technology*, vol. 4, pp. 9–21, July 2014.

[17] N. Li, L. Jia, and P. Zhang, "Detection and volume estimation of bubbles in blood circuit of hemodialysis by morphological image processing," in *2015 IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, pp. 228–231, July 2015. ISSN: 2326-8239.

[18] K. A. Mat Said, A. Jambek, and N. Sulaiman, "A study of image processing using morphological opening and closing processes," *International Journal of Control Theory and Applications*, vol. 9, pp. 15–21, Jan. 2016.

[19] S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*. USA: California Technical Publishing, 1997.

[20] L. Vincent, "Morphological Area Openings and Closings for Grey-scale Images," in *Shape in Picture* (Y.-L. O, A. Toet, D. Foster, H. J. A. M. Heijmans, and

P. Meer, eds.), NATO ASI Series, (Berlin, Heidelberg), pp. 197–208, Springer, 1994.

[21] P. Ganesan and G. Sajiv, "A comprehensive study of edge detection for image processing applications," in *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, pp. 1–6, Mar. 2017.

[22] D. Ziou and S. Tabbone, "'Edge detection techniques: An overview'," *International Journal of Patttern Recognition and Image Analysis*, vol. 4, pp. 537–559, Jan. 1998.

[23] P. Ganesan, V. Rajini, and R. I. Rajkumar, "Segmentation and edge detection of color images using CIELAB color space and edge detectors," in *INTERACT-2010*, pp. 393–397, Dec. 2010.

[24] S. Wang, F. Ge, and T. Liu, "Evaluating Edge Detection through Boundary Detection," *EURASIP Journal on Advances in Signal Processing*, vol. 2006, pp. 1–15, Dec. 2006. Number: 1 Publisher: SpringerOpen.

[25] I. Sobel, "An Isotropic 3x3 Image Gradient Operator," *Presentation at Stanford A.I. Project 1968*, 2014.

[26] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, pp. 679–698, Nov. 1986. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[27] M. Heath, S. Sarkar, T. Sanocki, and K. Bowyer, "A robust visual method for assessing the relative performance of edge-detection algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 1338–1359, Dec.

1997. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

[28] P. V. Hough, "Machine analysis of bubble chamber pictures," in *Proc. of the International Conference on High Energy Accelerators and Instrumentation, Sept. 1959*, pp. 554–556, 1959.

[29] J. Illingworth and J. Kittler, "A survey of the hough transform," *Computer Vision, Graphics, and Image Processing*, vol. 44, pp. 87–116, Oct. 1988.

[30] R. O. Duda and P. E. Hart, "Use of the Hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, pp. 11–15, Jan. 1972.

[31] A. Herout, M. Dubská, and J. Havel, "Review of Hough Transform for Line Detection," in *Real-Time Detection of Lines and Grids: By PClines and Other Approaches* (A. Herout, M. Dubská, and J. Havel, eds.), SpringerBriefs in Computer Science, pp. 3–16, London: Springer, 2013.

[32] Y. parsa, H. Hosseinzadeh, and M. Effatparvar, "Development Hough transform to detect straight lines using pre-processing filter," *International Journal of Information, Security and Systems Management*, vol. 4, no. 2, pp. 448–456, 2015. Publisher: Islamic Azad University E-Campus _eprint: https://ijissm.ec.iau.ir/article_559196_275d2eb157f31678307c6a963a54ee34.pdf.

[33] V. Chandola, "Anomaly Detection : A Survey," *ACM Computing Surveys*, 2009.

[34] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, and M. Kloft, "Deep Semi-Supervised Anomaly Detection," Feb. 2020. arXiv:1906.02694 [cs, stat].

[35] F. Ye, H. Zheng, C. Huang, and Y. Zhang, "Deep Unsupervised Image Anomaly Detection: An Information Theoretic Framework," in *2021 IEEE International Conference on Image Processing (ICIP)*, pp. 1609–1613, Sept. 2021. ISSN: 2381-8549.

[36] R. Chalapathy and S. Chawla, "Deep Learning for Anomaly Detection: A Survey," Jan. 2019. arXiv:1901.03407 [cs, stat].

[37] F. Ulger, S. E. Yuksel, and A. Yilmaz, "Anomaly Detection for Solder Joints Using β-VAE," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 11, pp. 2214–2221, Dec. 2021. arXiv:2104.11927 [cs].

[38] D.-M. Tsai and P.-H. Jen, "Autoencoder-based anomaly detection for surface defect inspection," *Advanced Engineering Informatics*, vol. 48, p. 101272, Apr. 2021.

[39] J. K. Chow, Z. Su, J. Wu, P. S. Tan, X. Mao, and Y. H. Wang, "Anomaly detection of defects on concrete structures with the convolutional autoencoder," *Advanced Engineering Informatics*, vol. 45, p. 101105, Aug. 2020.

[40] R. Harik, C. Saidy, S. J. Williams, Z. Gurdal, and B. Grimsley, "Automated Fiber Placement Defect Identity Cards: Cause, Anticipation, Existence, Significance, and Progression," in *SAMPE 2018*, (Long Beach, CA), SAMPE 2018, May 2018. NTRS Author Affiliations: University of South Carolina, NASA Langley Research Center NTRS Report/Patent Number: NF1676L-29045 NTRS Document ID: 20200002536 NTRS Research Center: Langley Research Center (LaRC).

[41] E. Oromiehie, B. G. Prusty, P. Compston, and G. Rajan, "Automated fibre placement based composite structures: Review on the defects, impacts and inspections techniques," *Composite Structures*, vol. 224, p. 110987, Sept. 2019.

[42] K. Fayazbakhsh, M. Arian Nik, D. Pasini, and L. Lessard, "Defect layer method to capture effect of gaps and overlaps in variable stiffness laminates made by Automated Fiber Placement," *Composite Structures*, vol. 97, pp. 245–251, Mar. 2013.

[43] M. Lan, D. Cartié, P. Davies, and C. Baley, "Influence of embedded gap and overlap fiber placement defects on the microstructure and shear and compression properties of carbon–epoxy laminates," *Composites Part A: Applied Science and Manufacturing*, vol. 82, pp. 198–207, Mar. 2016.

[44] S. Gholizadeh, "A review of non-destructive testing methods of composite materials," *Procedia Structural Integrity*, vol. 1, pp. 50–57, Jan. 2016.

[45] F. Shadmehri, O. Ioachim, O. Pahud, J.-E. Brunel, A. Landry, V. Hoa, and M. Hojjati, "LASER-VISION INSPECTION SYSTEM FOR AUTOMATED FIBER PLACEMENT (AFP) PROCESS," *20th International Conference on Composite Materials*, 2015.

[46] P. D. Juarez and E. D. Gregory, "In Situ Thermal Inspection of Automated Fiber Placement for manufacturing induced defects," *Composites Part B: Engineering*, vol. 220, p. 109002, Sept. 2021.

[47] B. Denkena, C. Schmidt, K. Völtzer, and T. Hocke, "Thermographic online monitoring system for Automated Fiber Placement processes," *Composites Part B: Engineering*, vol. 97, pp. 239–243, July 2016.

[48] J. Cemenska, T. Rudberg, and M. Henscheid, "Automated In-Process Inspection System for AFP Machines," *SAE International Journal of Aerospace*, vol. 8, pp. 303–309, Sept. 2015.

[49] D. Maass, "Progress in automated ply inspection of AFP layups," *Reinforced Plastics*, vol. 59, pp. 242–245, Sept. 2015.

[50] S. Roy, M. Palardy-sim, M. Rivard, G. Lamouche, C. Padioleau, A. Yousefpour, G. Lund, M. Zupan, M. Klakken, S. Albers, and R. Harper, "AUTOMATED FIBER PLACEMENT INSPECTION : ENABLING A PARADIGM SHIFT IN QUALITY CONTROL TOWARDS HIGH-FIDELITY SURFACE PROFILOMETRY," *The Composites and Advanced Materials Expo*, 2019.

[51] C. Sacco, A. Baz Radwan, A. Anderson, R. Harik, and E. Gregory, "Machine learning in composites manufacturing: A case study of Automated Fiber Placement inspection," *Composite Structures*, vol. 250, p. 112514, Oct. 2020.

[52] C. Schmidt, T. Hocke, and B. Denkena, "Deep learning-based classification of production defects in automated-fiber-placement processes," *Production Engineering*, vol. 13, pp. 501–509, June 2019.

[53] B. Wei, K. Hao, X.-s. Tang, and L. Ren, "Fabric Defect Detection Based on Faster RCNN," in *Artificial Intelligence on Fashion and Textiles* (W. K. Wong, ed.), Advances in Intelligent Systems and Computing, (Cham), pp. 45–51, Springer International Publishing, 2019.

[54] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," Oct. 2014. arXiv:1311.2524 [cs].

[55] X. Wang, Y. Gao, J. Dong, X. Qin, L. Qi, H. Ma, and J. Liu, "Surface defects detection of paper dish based on Mask R-CNN," *SPIE*, vol. 10828, p. 108280S, July 2018. Conference Name: Third International Workshop on Pattern Recognition ADS Bibcode: 2018SPIE10828E..0SW.

[56] M. Rivard, M. Palardy-Sim, G. Lamouche, S. Roy, C. Padioleau, A. Beauchesne, D. Levesque, F. Boismenu, L. G. Dicaire, J. Boisvert, S. Peters, J. Chen, M. A. Octeau, J. B. Robles, F. Ferland, M. Tanguay, J. Hissett, D. Swope, S. Albers, R. Harper, K. Wright, B. Buhrkuhl, M. Klakken, G. Lund, and A. Yousefpour, "Enabling responsive real-time inspection of the automated fiber placement process," *International SAMPE Technical Conference*, vol. 2020-June, 2020.

[57] M. Palardy-Sim, M. Rivard, G. Lamouche, S. Roy, C. Padioleau, A. Beauchesne, D. Levesque, L.-G. Dicaire, J. Boisvert, S. Peters, and others, "Advances in a Next Generation Measurement & Inspection System for Automated Fibre Placement," in *Proceedings of the Manufacturing & Processing Technologies Conference in the Composites and Advanced Materials Expo (CAMX), Anaheim, CA, USA*, pp. 23–26, 2019.

[58] J. Azzeh, B. Zahran, and Z. Alqadi, "Salt and pepper noise: Effects and removal," *JOIV: International Journal on Informatics Visualization*, vol. 2, no. 4, pp. 252–256, 2018.

[59] G. Deng and L. Cahill, "An adaptive Gaussian filter for noise reduction and edge detection," in *1993 IEEE Conference Record Nuclear Science Symposium and Medical Imaging Conference*, pp. 1615–1619 vol.3, 1993.

[60] L. He, X. Ren, Q. Gao, X. Zhao, B. Yao, and Y. Chao, "The Connected-Component Labeling Problem: A Review of State-of-the-Art Algorithms," *Pattern Recognition*, vol. 70, Apr. 2017.

[61] H. Greiner, "A survey on univariate data interpolation and approximation by splines of given shape," *Mathematical and Computer Modelling*, vol. 15, pp. 97–106, Jan. 1991.

[62] 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)*, vol. 1, pp. 886–893 vol. 1, 2005.

[63] A. Danker and A. Rosenfeld, "Blob Detection by Relaxation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-3, pp. 79–92, 1981.

[64] H. Kong, H. C. Akakin, and S. E. Sarma, "A generalized Laplacian of Gaussian filter for blob detection and its applications," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1719–1733, 2013. Publisher: IEEE.

[65] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision 2004 60:2*, vol. 60, pp. 91–110, Nov. 2004. Publisher: Springer.

[66] A. Ghamisi, T. Charter, L. Ji, M. Rivard, G. Lund, and H. Najjaran, "Anomaly Detection in Automated Fibre Placement: Learning with Data Limitations," July 2023. arXiv:2307.07893 [cs, eess].