University of Hamburg

University of L'Aquila

Joint Master's Programme - MathMods
Mathematical Modelling in Engineering: Theory, Numerics, Applications

Master of Science	**Laurea Magistrale**
Mathematical Modelling in Engineering	**Mathematical Modelling**
UNIVERSITY OF HAMBURG (UHH)	**UNIVERSITY OF L'AQUILA (UAQ)**

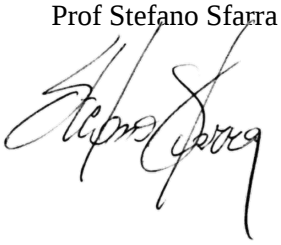
Master's Thesis

A new approach for automatic defect detection via thermal image processing and deep learning tools

Supervisor
Prof Antonio Cicone

Candidate
Stephen Naboth

Co-advisor
Prof Stefano Sfarra



Student ID (UAQ): 274443
Student ID (UHH): 7479127

Academic Year 2021/2022

Acknowledgements

This thesis would not have been possible without the support of many people. Many thanks to my advisers, Prof. Antonio Cicone and Prof. Stefano Sfarra for their continuous guidance, tremendous and invaluable support, patience, and strong encouragement to me throughout the entire journey of research. Thanks to the Mathmods program for providing me with the financial support and opportunity to complete this project. And finally, my parents and numerous friends who endured this long process with me always offered support and love. May GOD bless you all.

Abstract

This master thesis work focuses on automatic defect identification and segmentation by a combination of signal processing and deep learning algorithms in infrared thermography (IRT). IRT is a type of Non-destructive evaluation technique that involves identifying structural damage in an object of interest without applying any permanent damage and modification. The data used were thermal images recorded by the infrared camera are 2D non-stationary signals. First, data pre-processing is applied to identify and detect defects to clean the images. We briefly introduce and discuss the decomposition of signal methods; then, for this thesis Fast Iterative Filtering (FIF), a decomposition method, is used as the pre-processing algorithm due to its ability to handle non-stationary signals. Then, a deep learning method is applied to detect the defect. Deep learning has made remarkable progress in image processing applications in recent years. Due to insufficient training data, deep learning algorithms remain mainly unexplored in this research direction, and only a few publications involve their application to IRT. We introduce the Deep learning algorithm Mask Region Based Convolutional Neural Networks (RCNN) and how it is coupled with IRT to identify and classify the defect.

Contents

List of Figures

CHAPTER 1

Introduction

1.1 Overview

In this chapter, we introduce the thesis background and focus area. We highlighted the motivations, scope, and thesis organization. We will explore the Non-destructive evaluation (NDE) method in defect detection. Infrared cameras were used for data acquisition of thermal images which can be reordered as nonstationary 2D signals.

Signal Processing techniques are described and justified for pre-processing thermal images to remove noise recorded by the infrared camera. In this thesis, Fast Iterative Filtering (FIF) is applied in the pre-processing of thermal images.

A deep Learning approach is proposed in automatically recognizing pixel-level defects and creating bounding boxes around the thermal images. Mask RCNN is used thanks to its ability in segmenting the defects. Artificial neural network theory is briefly explained, and Mask RCNN architecture is described and applied.

1.2 Importance of Non-Destructive Evaluation

Non-destructive Evaluation (NDE) [3] is a group of techniques used to analyze and evaluate a material property without causing damage. NDE methods rely upon electromagnetic radiation, sound and other signal conversions to examine a wide variety of objects [5]. Infrared Thermography (IRT) is one of many NDE techniques used to "see the unseen".

Inspecting materials and products using IRT have been widely applied to evaluate subsurface defects, hidden structures, etc. IRT is based on imaging an object using an infrared camera by sensing the infrared (thermal) radiation emitted by the object for the quality control of materials such as metals and composites. Due to thermal wave propagation and attenuation, each heterogeneous thermal element perturbs the thermal wave propagation on the object surface compared to the surrounding area of study.

The infrared camera records this thermal perturbation, and the results can be analyzed to obtain further information. This approach has some advantages, such as low cost since no contact and damage of the object is performed. In addition, safety is an advantage since some objects, such as hot or toxic materials are not in contact with the inspector, and some objects can be out of reach to contact. Also, some objects, such as historical paintings, can be delicate to touch; thus, avoiding damage is a priority.

The main objective of the IRT technique in this thesis is to detect and classify the Regions of Interest (ROIs) that could be represented as an anomaly or a defect by analyzing the sequence of images.

1.3 Pre-processing techniques

Pre-processing is generally used to correct some parts of thermal images affected by, for example, dead pixels and vignetting. Pre-processing methods improve defect visibility during data processing in infrared thermography by cleaning the thermal images [29]. These methods include Empirical Mode Decomposition (EMD), Discrete Wavelet Transform (DWT), and Iterative filtering (IF).

In 1998 [42], Huang and his research group at NASA devised the so-called EMD algorithm to handle nonstationary signals by decomposing the signal into simple components which contain a unique instantaneous frequency at each instant of time. The relevant components based on the focus of the study were identified and combined to form the clean signal. A few years later, Huang and his group devised the so-called Ensemble Empirical Mode Decomposition (EEMD) algorithm, which allowed them to overcome the instability of the original EMD algorithm.

The success of the two algorithms inspired researchers to work on alternative methods for decomposing a signal into a few simple and meaningful components. One of them, called the IF method, and its faster alternative, the Fast Iterative Filtering (FIF) algorithm, are based on iterations like EMD and EEMD, that is, no assumptions are required on the kind of signal we want to decompose. In this thesis, we adopted the 2D FIF as our pre-processing tool for the 2D thermal images to improve performance when applying post-processing approaches.

1.4 Deep learning technique

Traditionally, regular pattern-based unsupervised data processing has been used for quite a long time. It led researchers to seek innovative methods to enhance defects' visibility further and analyze the defect characterization in industrial applications, which are still crucial issues in thermographic NDE literature [33]. Regular human inspection of structures, machines, and materials can easily be hampered by fatigue and human judgment error. Therefore, an automated inspection system would be ideal since it is compatible with high inspection rates to avoid human inspector fatigue, to meet accuracy and quality demands for quantitative analysis, and maintenance costs are highly recommended in the NDE of IRT.

This study proposes deep learning algorithms to achieve automatic defect detection and precise localization (subsurface defects case) from different thermal image sequences. Since the beginning of the twenty-first century, the deep learning [47] (deep neural network)-one of the fields of artificial intelligence, has made substantial progress. Deep learning neural networks have shown their capability to outperform most of the other approaches existing previously in a significant number of applications. It has contributed to making them very popular in diverse scientific communities.

However, one must consider the computational expense caused by multilayers of deep learning. The requirement for training datasets to be labeled and used

in the network remains unexplored in NDE projects. Few works in the literature deal with defect detection in passive infrared thermography coupling with deep learning algorithms due to training complexity and training data limitation. As a result, innovative approaches need to be developed to further improve the feasibility of deep Learning models with infrared thermography.

Compared to the typical neural network, deep learning is making huge strides in solving problems that were difficult to handle in the artificial intelligence research fields for many years. As a result, it has been applied to different research fields. For example, computer vision has surpassed other learning techniques in predicting and analyzing the activity based on scientific data, especially in image classification and image recognition.

This thesis introduces the object segmentation method for defect segmentation and identification of each object at the pixel level , and, in particular, the deep learning algorithm Mask RCNN is presented. Mask RCNN is a conceptually flexible, simple, and general framework for object instance segmentation.

As we will see, detailed and enough parameters to train data in the data trained model can lead to better performance for the trained results. However, collecting a considerable amount of data is computationally expensive in training. Therefore, in this thesis, we collected a small amount of thermal data due to the constraints caused by difficulty in getting the license, the long time required to collect the data, and the limitation of camera storage capacity. Due to small data, transfer learning will be applied to train the deep spatial characteristic models to identify or segment defects in this research.

We use deep neuron networks to extract more valuable data to train a learning system using artificial intelligence learning methods. The research on infrared thermography for defect detection using deep learning algorithms is a relatively new topic in the field of NDE.

For this project, we propose to design an advanced deep learning algorithm to extract more information from IRT for NDE to achieve infrared defects visibility enhancement and automatic defect detection and segmentation. Furthermore, the proposed method allows to improve the detection of more profound subsurface defects for limited IRT signals.

1.5 Literature review

In [17] deep learning approaches were proposed on infrared thermography inspection, intended to identify defects efficiently and accurately automatically. Deep learning algorithms were applied to achieve automatic defect detection and precise localization (subsurface defects case) from different thermal image sequences. The algorithms applied included Mask-RCNN, Yolo-v3 and Res-unet. Mask RCNN was applied for instance segmentation, Res-u-net was applied for semantic segmentation, and YOLO-V3 was applied for object detection. Their performance was evaluated and compared. These mentioned methods showed remarkable efficacy in improving defect visibility during NDE. The proposed methods achieved effectively the defect identification and segmentation with limited data.

Work devoted to the fast iterative filtering performance in pre-processing thermographic data was presented in [29]. FIF2 algorithm brief theory was studied, and its performance as a pre-processing tool was done. FIF2 was

introduced, discussed, and compared to previously developed techniques. Pre-processed data were further analyzed during the post-processing step, demonstrating the reliability of FIF2 in enhancing thermal imprints, which leads to improved detection of subsurface features. In particular, enhanced thermal imprints highlight the shape of the grid of glass fibres present beneath an external coating of hemp fibres (and, in general, added to the whole specimen along the x-y vectors). Finally, a validation among numerical and experimental (thermographic) data is provided thanks to the Parker method.

In [19] infrared thermography is used for NDE of fibre reinforced polymer (FRP) rehabilitated structures. Infrared thermography presented a potentially efficient non-contact, real-time inspection and data interpretation technique. The technique was reviewed while assessing the variations thereof. Also, critical assessment of challenges must be overcome before the method can be routinely used to inspect externally bonded FRP composites.

In [12] multidimensional iterative filtering decomposition was applied to detect Hyperspectral chemicals. Chemical plumes from natural or anthropogenic atmospheric emissions can be unexpected and toxic. The detection and classification of such plumes in an efficient way would reduce the risk of harmful exposures. IRT was used as the pre-processing algorithm for a hyperspectral dataset before classification. It allowed the hypercubes to be decorrelated and mean-centred, thus, proving to be an equivalent procedure to the mean-centring and whitening of the data.

In [44] deep learning is applied for automatic defect detection and segmentation of tunnel surface using modified Mask RCNN. Safety of tunnel, periodic inspection to detect surface defects. Traditionally it relies on using "naked eyes" inspection. However, these methods with manual feature extraction do not perform well in detecting tunnel defects due to the complicated background of tunnel surfaces. The work proposes a deep learning algorithm called Mask RCNN to address this problem and improve object detection accuracy.

Lastly, in work titled “deep learning for infrared thermal image-based machine health monitoring” [22], deep learning was applied to automatically identify the condition of a machine by creating and classifying features that summarize the characteristics of measured signals. Deep learning was applied to the infrared thermal video to determine the machine’s condition automatically. The requirements to be detected were oil level prediction and machine fault detection. The system could detect the conditions very accurately (i.e. 91.67% and 95% accuracy of the individual use cases). The proposed approach can detect many conditions in rotating machinery very accurately without requiring any detailed knowledge about the underlying physics, thus having the potential to simplify condition monitoring using complex sensor data significantly. Furthermore, essential regions in the infrared thermal images can be identified related to specific conditions, potentially leading to new physical insights.

1.6 Summary of the work

The thesis continues as follows; in the following chapter, we review signal decomposition methods for signal pre-processing applications. In particular,

EMD and FIF approaches are defined and compared, and characteristics and examples are demonstrated. Chapter 3 is dedicated to Machine Learning, focusing on the deep learning branch and their possible application to IRT. The fourth chapter reviews the numerical applications of the Fast Iterative Filtering method and Mask RCNN algorithm. In Chapter 5, we make some recommendations and arrive at conclusions of the present work.

PART I

The First Part

CHAPTER 2

Signal Decomposition

Nonstationary signals, which are signals whose frequency and amplitude are not constant, are prevalent in the real world. Their solution provides exceptional discoveries in many fields of research [18]. Consider, for instance, the atmospheric temperature of the globe recorded for instance daily temperature for several years. In economics, the record of inflation rate for some time. In Medicine, biological signals reflect the presence signifies the ionic and electrical activities of the muscular and the neural cells in a synchronized manner [24]. In geophysics, analyzing signal anomalies in Schumann Resonances that occur days before a major seismic wave [26]. In computer vision, with application to signal image video process and multi-scale image fusion [1]. Several signal processing methods have been proposed, such as Short Time Fourier Transform (STFT), Wavelet Transform (WT), and derived methods [9]. STFT assumes stationarity within a short time interval for a given window. Thus, the performance is poor for a signal with abundant fast-changing frequencies common in nature. However, the wavelet transform is updated also essentially a window tunable Fourier transform that did not get rid of the limitations of the Fourier transform and the wavelet basis is difficult to select, which lacks adaptability [18]. Furthermore, both processes have predetermined bases making them not suitable for data-driven analysis. Thus, there is a need for methods that are more accurate in the time-frequency representation.

2.1 Empirical Mode Decomposition

In 1998, Huang and his research group at NASA developed the Empirical Mode Decomposition (EMD) algorithm [18] [30]. This addresses the question of decomposing non-stationary signal into simpler components without prior knowledge of the basis. The components are named the Intrinsic Mode Functions(IMFs). IMF is a function where the number of extrema and zero crossings must either be equal or differ by only one and the mean value of the envelope defined by the local maxima and minima is zero [4] [35].

The IMF has to be extracted properly from signal in order to keep its properties and physical meaning of signal unaltered. Significant analysis is done using the IMFs as they bring the meaningfulness of the signal [35]. The key process behind this technique is the sifting process that extracts the highest oscillations of a signal f by subtracting its moving average $M(f)$ from the signal [43].

Let M be the operator that captures the moving average of the signal f, and let:

$$f_1 = S_{1,1}(f)(t) = f(t) - M(f)(t) \tag{2.1}$$

where f_1 be the fluctuating part. Iterating leads to compute:

$$f_n = S_{1,n-1}(f_{n-1})(t) = f_{n-1} - M(f_{n-1})(t) \tag{2.2}$$

The first IMF denoted by $IMF_1(t)$ is given by

$$IMF_1 = \lim_{n\to\infty} S_{1,n}(f) \tag{2.3}$$

where S subscripts 1 and n denote the first IMF and $n-th$ iteration respectively[30]. When the limit is achieved the moving average of IMF_1 is the zero function. For $m \geq 1$ IMFs have been determined, the the residual is given by:

$$r(t) = f(t) - \sum_{j=1}^{m} IMF_j(t) \tag{2.4}$$

The iteration stops when r becomes a trend signal, that is, with atmost one local maxima or minima. Rearranging the equation, the signal data set is decomposed as,

$$f(t) = \sum_{j=1}^{m} IMF_j(t) + r(t) \tag{2.5}$$

The moving average is the mean of two envelops, upper and lower, that are cubic spines connecting local maxima and local minima.

Upon further study the EMD was found to be sensitive to small pertubations. Thus small pertubation especially white noise lead to completely different decomposition. To overcome this Huang, introduced the Ensembled Empirical Mode Decomposition (EEMD)[25]. It involved adding of white noises realization then several decompositions computed as the mean of many different trials produced using the (EMD). Thus achieving robustness. Several alternative methods have been derived. Among them, the only one based on iterations is the Iterative Filtering (IF).

2.2 Iterative Filtering

Lin et al in 2009, proposed an alternative method called Iterative Filtering[9]. This method addresses the issues related to EMD such as stability and convergence of sifting process. This technique has the same structure of the EMD. The difference is, it replaces the moving average with convolution of the given signal with filters, for instance, a double average filter.

Definition 2.2.1. A function $\omega : [-k, k] \to \mathbb{R}$ is a filter if it is nonnegative, bounded, even, continuous, and $\int_{\mathbb{R}} \omega(t)\, dt = 1$[10]. Also, a double convolution filter ω is the self-convolution of a filter ϖ, that is $\omega = \varpi * \varpi$. The size of the filter is half of its support: filter length of

$$\omega = \frac{1}{2} m\{t : \omega(t) > 0\}.$$

The filter function is used for signal denoising. Filter methods are able to handle texture, edges, and high frequency signals all at once. When applied to images, these methods are able to separate the majority of the signal from the noise[3]. This is achieved by convolution of the signal weighted using a filter w_n with non-zero values concentrated on a finite interval $[-k, k]$[9]:

$$M(f_n)(x) = \int_{-k}^{k} f_n(x+t)w_n(t)\,dx \tag{2.6}$$

where k_n is the filter length. Define $f_1 = f$ to capture the flactuation f_n, through the operator:

$$f_{n+1} = f_n - M_n(f_n) \tag{2.7}$$

where the subscript is the number of IMFs. The IF stops when the reminder r is a trend. Hence, the signal is decomposed as in the EMD[43]. There are two loops, an inner loop and an outer loop in the iterative filtering algorithm. The inner loop is to compute and capture a single IMF and the outer loop is to derive all the IMFs of the original signal. Combined with the Fast Fourier Transform(FFT), it formed an algorithm called Fast Iterative Filtering (FIF).

Algorithm 1 Iterative Filtering $IMF = IF(f)$

```
IMF = {}
while the number of extrema of f ≥ 2 do
    f_1 = f
    while the stopping criterion is not satisfied do
        Compute filter length l_m for f_m(x)
        f_{m+1}(x) = f_m(x) − ∫_{−k_m}^{k_m} f_m(x + t)w_m(t)dt
        m = m + 1
    end while
    IMF = IMF ∪ {f_m}
    f = f − f_m
end while
IMF = IMF ∪ {f}
```

The length k_n can be computed as follow

$$k_n = 2\left[v\frac{N}{z}\right] \tag{2.8}$$

where v is set as 1.6. N is the total number sample parts of original signal. z is the number of its extreme points.

Theorem 2.2.2 (Convergence of the Iterative Filtering method[14])**.** *Given the filter function $w(t)$, $t \in [-k, k]$, be L^2, symmetric, nonnegative, $\int_{-k}^{k} w(t)dt = 1$, and let $f(x) \in L^2$. If $\hat{w}(t) = 0$ or $|1 - \hat{w}(t)| < 1$, where $\hat{w}(t)$ os the Fourier transform of $w(t)$ computed at frequency ξ, Then $M^n(f)(x)$ converges and;*

$$IMF_1 = \lim_{n\to\infty} M^n(f)(x) = \int_{-\infty}^{\infty} \hat{f}(x)\chi_{\hat{w}(\xi)=0}e^{2\pi\xi x}\,d\xi \tag{2.9}$$

Thus the method converges for sure to the limit function (2.9) which depends only on the shape of the filter function chosen and the support length selected by the method (2.8). The theorem provides sufficient conditions on the filter that guarantee the convergence of the inner loop[11]. The sufficient conditions are not unrealistic.

For example, let the double average filter $a(t)$ that satisfies the sufficient conditions be given by

$$a(t) = \frac{k+1-|t|}{(k+1)^2}, \quad t \in [-k, k]$$

where $\mathcal{F}(a)(\xi) = 0$ when $\xi = \frac{l}{k+1}$, $1 \leq l \leq k+1$. In most cases, it is easy to obtain filters with the property $\mathcal{F}(\omega)(\xi) = 0$ or $|1 - \mathcal{F}(\omega)(\xi)| < 1$. Thus, it is known that for nonnegative and symmetric filters ω, $\mathcal{F}(\omega)(\xi)$ is real and

$$\mathcal{F}(\omega)(\xi) = \int_{-\infty}^{\infty} \omega(t) cos(-2\pi i t\xi)\, dt$$

Since $\int_{\mathbb{R}} \omega(t)\, dt = 1$, the we have

$$\begin{aligned} |\mathcal{F}(\omega)(\xi)| &= \left| \int_{-\infty}^{\infty} w(t) \cos(-2\pi i t\xi) \mathrm{d}t \right| \leq \int_{-\infty}^{\infty} |w(t) \cos(-2\pi i t\xi)|\, \mathrm{d}t \\ &< \int_{-\infty}^{\infty} |w(t)|\, \mathrm{d}t = \int_{-k}^{k} w(t) \mathrm{d}t = 1 \end{aligned} \tag{2.10}$$

Hence for a nonnegative and symmetric filters ω, $t \in [-k, k]$, $-1 < \mathcal{F}(\omega)(\xi) < 1$, for every $\xi \in \mathbb{R}$. For $0 \leq \mathcal{F}(\omega)(\xi) < 1$, for all $\xi \in \mathbb{R}$, consider filter $v(t)$, $t \in [-2k, 2k]$ given by the convolution of the filter ω, $t \in [-k, k]$, with itself. Given by

$$v(t) = \omega(t) * \omega(t).$$

The Fourier transform of $v(t)$ is given by $\mathcal{F}(v)(\xi) = \mathcal{F}(\omega)(\xi) * \mathcal{F}(\omega)(\xi)$ satisfying the condition $0 \leq \mathcal{F}(v)(\xi) < 1$, for all $\xi \in \mathbb{R}$. Thus, every filter defined by the convolution of a nonnegative, symmetric, and finitely supported L^2 filter with itself satisfies the sufficient conditions of Theorem 2.2.2.

2.3 Numerical Examples

Example 1

Figure 2.1 is a signal to be decomposed by the Iterative filtering algorithm. IF produces decomposition plotted in figure 2.2, where we report an IMF and the trend compared to the ground truth. The signal is composed by adding two functions:

$$y_1 = (2(t-0.5)^2 + 0.2) sin(20\pi t + 0.2 cos(40\pi t)) \tag{2.11}$$

$$y_2 = 5(t-0.5)^2 \tag{2.12}$$

Figure 2.1: Artificial Signal generated

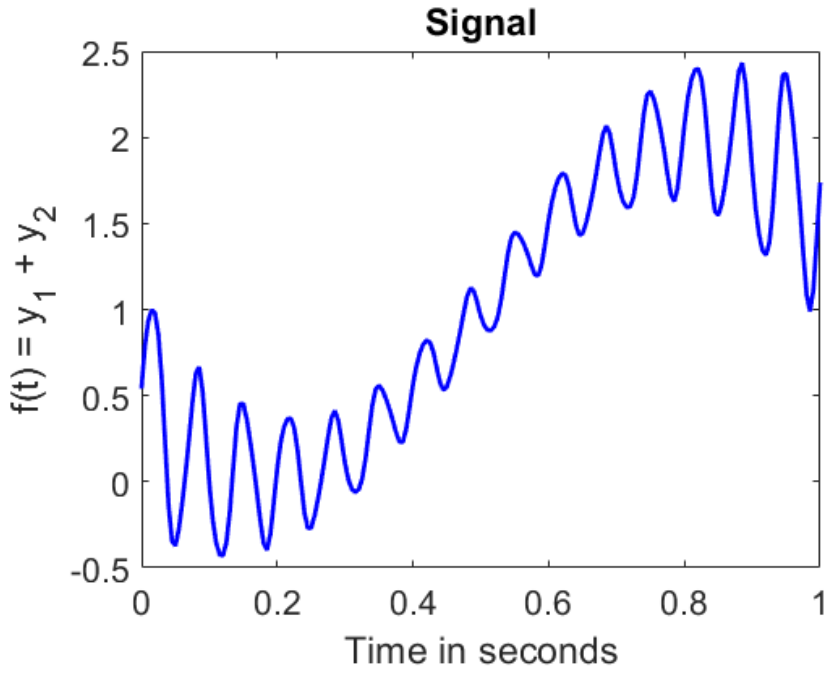
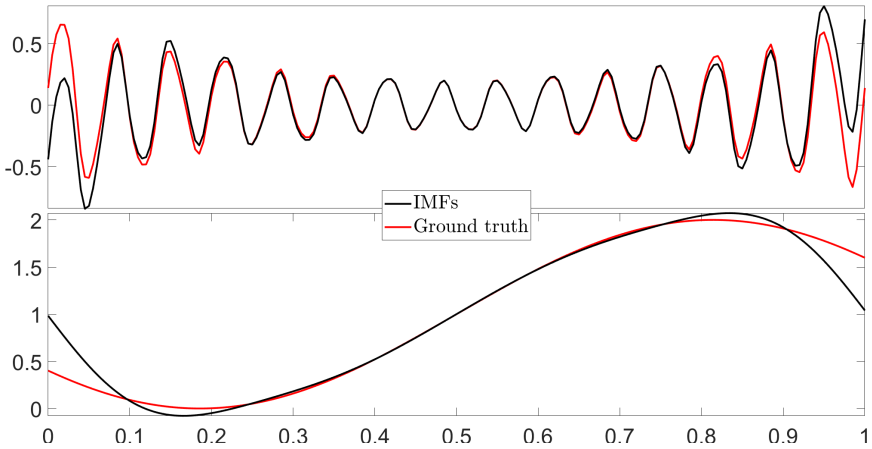


Figure 2.2: Decomposition of non-stationary signal using IF algorithm



In Figure 2.2 the red curves represent the original components (ground truth) of the signal, whereas the black curves represent the intrinsic mode functions (IMFs) computed by the IF algorithm. From Figure 2.2, there are evident errors nearby the boundaries when decomposing the signal by IF algorithm. Boundary conditions influence the decomposition of a signal to an extent that increases with the component scale. The boundary was pre-extended periodically and

anti-symmetrically a method assumed to be optimal [30]. Extensions should be properly handled, for robust decomposition of the original signal. However, based on Figure 2.2, the pre-extension is not optimal since near the boundary where the frequency and amplitude change rapidly, the error increases. Thus, the amplitude and frequency rate of change affects the performance of the extension. The identification of an optimal extension is an open problem in the field for general signals. This problem is out of the scope of this work. From now on we assume that an optimal way of solving the problem can be identified case by case, and we leave the problem to future research.

Example 2

Here the IF is applied to Tsunami data recorded at Kawaihae, Hawaii, 4308hours from March 11, 2011 to September 13, 2011.The data is decomposed into several components as shown in Figure 2.4 where the first five IMFs represent the transient signals associated with the impact of the tsunami. The last three components instead reveal the basic wave height with its 'regular' patterns. The regularity of the patterns is not ideal due to several factors such as the pre-extension of the boundary and noise which arise from the error of the sensors [25].

Figure 2.3: Tsunami Signal

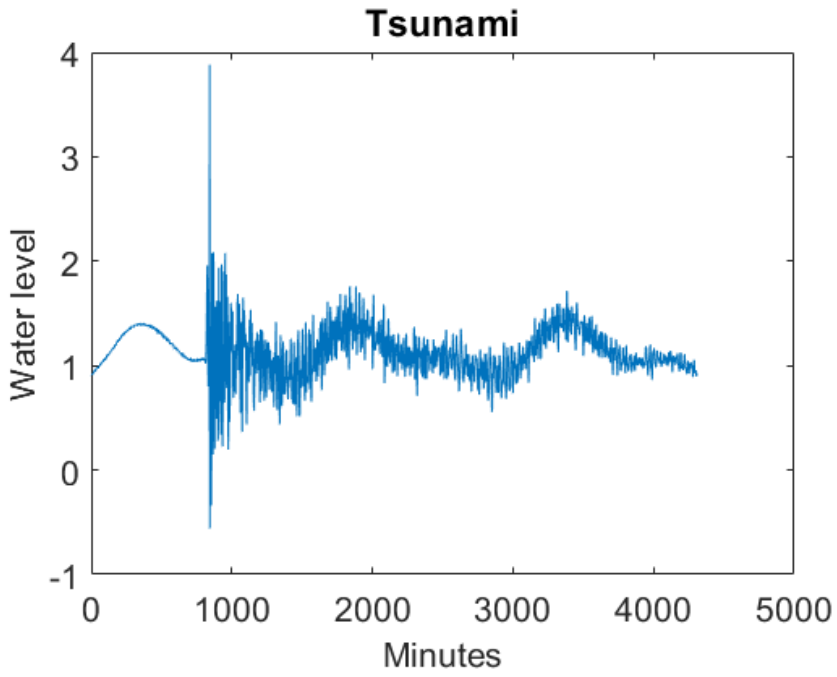
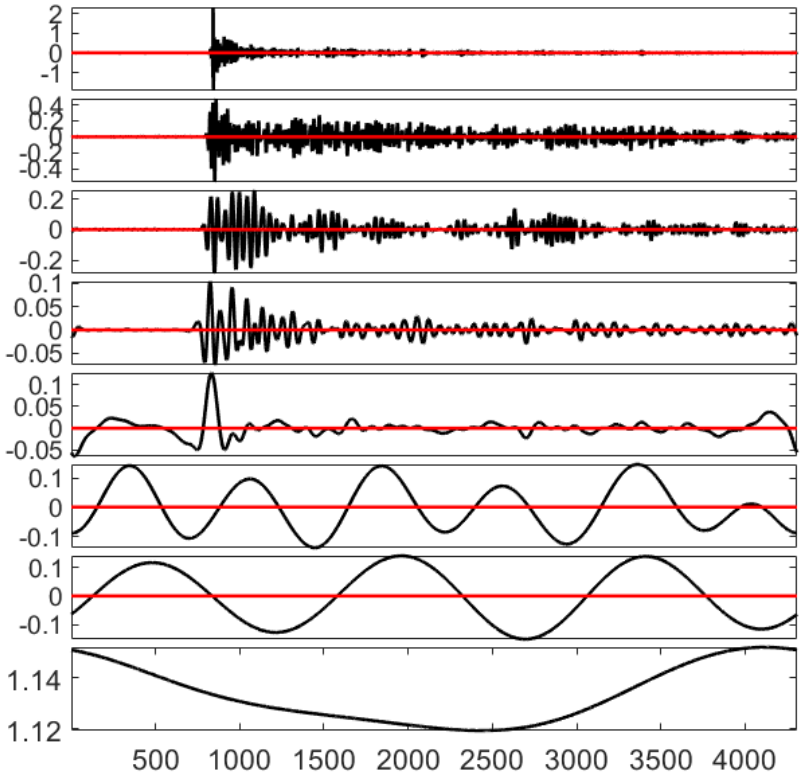


Figure 2.4 is a plot of the decomposed Tsunami signal. From the graph, the black signal is the decomposed signal with extended boundary, to make it periodic. While the red signal the boundary is not extended, the end effects are present. The decomposed signal had seven IMFs and a trend, where the

Figure 2.4: Decomposition of Tsunami signal using IF algorithm



first two IMFs represent the transient signals associated with the impact of the tsunami. The last four components instead reveal the basic wave height with its regular patterns with periods of approximately 12, 24, 36 and 72 hours. [11]

2.4 Multidimensional Iterative Filtering Method

Iterative Filtering can be extended to Higher dimensions in what is called Multidimensional iterative filtering (MIF) [13]. This involves the extension of IF code and its filter to higher dimensions. The MIF consists of two nested loops. The inner loop computes an IMF of an n-dimensional signals as the limit of a sequence generated taking the moving average and subtract it iteratively from the sequence [13]. The stopping criterion discontinues the iteration when it nears zero function. The outer loop updates the signal by subtracting from it the previously computed IMFs. The outer loop is iterated until the remainder is n-dimensional trend signal [9].

Algorithm 2 Multi-dimensional Iterative Filtering Algorithm $IMF = MIF(f)$

$IMFs = \{\}$
while the average number of extrema of $f \leq N_e$ **do**
  compute the filter support Ω for f
  $f_1 = f$
  while $f_{n+1}(x) = f_n - \int_\Omega f_n(x+t)w(t)d^k t$ **do**
    $f_{n+1}(x) = f_n(x) - \int_\Omega f_n(x+t)w(t)d^k t$
    $n = n + 1$
  end while
  $IMF = IMFs \cup \{f_n\}$
  $f = f - f_n$
end while
$IMF = IMFs \cup \{f_n\}$

The pseudo-code of MIF is provided in Algorithm 2, where f is k-dimensional signal to be decomposed, $\omega \in \mathbb{R}^n$ is a filter function with finite support $\Omega \subset \mathbb{R}^n$, $N_e \in \mathbb{N}$ is the number of local extrema allowed in the trend, usually set to two [13]. The condition for the filter ω are

$$\int_\omega w(t)\, d^k x \ = 1 \tag{2.13}$$

where ω is nonnegative L^2 function, finitely supported on $\omega \subset \mathbb{R}^n$. This implication does not mean all ω that satisfy the above property guarantee convergence of this technique. Thus careful chosen class of w should be applied.

In the one dimension IF algorithm, one dimension Fokker-Planck (FP) filters were applied as the weight function since they have the property of being infinitely smooth on the entire real line. The only drawback of such filters is that they are not known in an explicit form. However it is possible to compute them up to machine precision using numerical methods. Since it is infinitely differentiable, anywhere in the domain, they can be extended to k-dimension [37]. A 2D signal, applying 2D Fokker-plank partial differential equation. However the technique is computational expensive. Thus, as an alternative approach the tensor product can be considered. This method has the disadvantage that it does not ensure symmetry in the ordering where filters are applied in different directions.

A better approach is to make use of scaling, rotation and resampling of the one dimension filter to make the higher dimension version of it. For 2D, once the chosen one dimension Fokker-Planck filter ω has been computed, it can be extended to produce a numerical approximation $\tilde{\mathrm{W}}$ of the continuous two dimensional filter W, where the support $\Omega \subset \mathbb{R}^2$, which is bounded, connected and spherical set, is approximated by applying a lattice in $\mathbb{R}^2$. Consider a line $l_{h,k}$ for each point (h, k) on the lattice connected to the origin. Then W is approximated using values of $\tilde{\omega}$ along the line $l_{h,k}$. The process is repeated for every point inside the support Ω. Thus, the new filter $\tilde{W}$ obtained and scaled by its volume to ensure the property unit integral is stored. It follows that

$$\frac{1}{V_{\tilde{W}}} \int_\Omega \tilde{W}(t) \ d^k t \ = 1 \tag{2.14}$$

where $V_{\tilde{W}}$ is the volume of the new filter. The filter is produced using the previously proposed approach called Generalized Fokker-Planck (GFP) filter.

2.5 Convergence Result

Consider a 2 dimensional case, where the n-dimensional case follows from the result [9]. Define the Fourier Transform of a filter $\omega \in \mathbb{R}^2$ as

$$F(w)(\theta,\beta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} w(s,t)e^{-2\pi is\theta}e^{-2\pi is\beta}\,ds\,dt, (\theta,\beta) \in \mathbb{R}^2 \tag{2.15}$$

Define the operator

$$L(f)(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f((x,y)+(s,t))w(s,t)\,ds\,dt, \tag{2.16}$$

for $(\theta,\beta) \in \mathbb{R}^2$

since $s(f) := f - L(f) = (1-L)(f)$. Then, by linearity property of Fourier Transform

$$\mathcal{F}(S^n(t))(\theta,\beta) = \mathcal{F}((1-L)^n f)(\theta,\beta) \tag{2.17}$$

$$= [1-\mathcal{F}(w)(\theta,\beta)]^n\mathcal{F}(f)(\theta,\beta) \tag{2.18}$$

Theorem 2.5.1 ([13]). *Let $w(s,t),(s,t) \in \Omega$ be L^2, nonnegative, axial symmetric, satisfying*

$$\int_{\Omega} w(s,t)\,ds\,dt = 1 \tag{2.19}$$

and let $f(x,y) \in L^2(\mathbf{R}^2)$. If $|1-\mathcal{F}(w)(\theta,\beta)| \leq 1$ or $\mathcal{F}(w)(\theta,\beta) = 0$, Then $\{S^n(f)\}_{n\geq 1}$, converges and

$$\lim_{x\to\infty} S^n(f)(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathcal{F}(f)(\theta,\beta)\chi_{\{\mathcal{F}(w)(\theta,\beta)=0\}}e^{2\pi\theta x}e^{2\pi\beta y}\,ds\,dt \tag{2.20}$$

The proof is directly derived from the one dimensional case which can be found in [11].

The mathematical analysis of the IF method above allows both to guarantee its convergence and to accelerate the algorithm via the Fast Fourier Transform producing the aforementioned Fast Iterative Filtering(FIF) technique.Which is expanded to multi dimensions.

Fast Iterative Filtering

The properties of IF guarantee convergence, then accelerated via Fast Fourier Transform (FFT) to produce a new improved technique called Fast Iterative Filtering (FIF). The IF was extended to handle Multidimensional signals, thus, FIF is also extended to deal with Multivariate signals, that are multi-channelling and varying over space, but not over time [29].

In 2D Fast Iterative Filtering FIF2, the Pseudo code is provided in Algorithm 3, where f is the 2D signal, for example an image, to be decomposed, and the FFT_2 and $iFFT_2$ represents 2D Fast Fourier Transform and inverse Fast Fourier Transform respectively.

Algorithm 3 Multi-dimensional Iterative Filtering Algorithm IMF = FIF2(s)

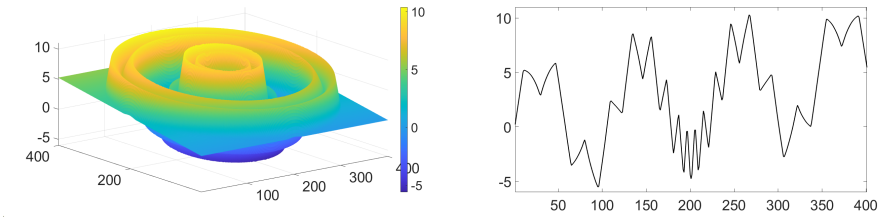
$IMFs = \{\}$
while the average number of extrema of $f \geq 2$ **do**
    compute the filter function w with support size L based on f
    $\tilde{w} = FFT_2(w)$
    $I_k = FFT_2(f)$
    while The stopping criterion is not satisfied **do**
        $I_k = (1 - \tilde{w})I_k$
    end while
    $IMF = IMFs \cup \{iFFT_2(I_k)\}$
    $f = f - I_k$
end while
$IMF = IMF \cup \{f\}$

This preprocessing method produces comparable results to the multi-dimensional ensemble empirical mode decomposition (MEEMD) for image preprocessing but at a faster rate [29]. MEEMD is an extension of the 1D EEMD algorithm to a signal encompassing multiple dimensions [42]. The experiment was done 2D images.

2.6 Numerical Examples

Example 1

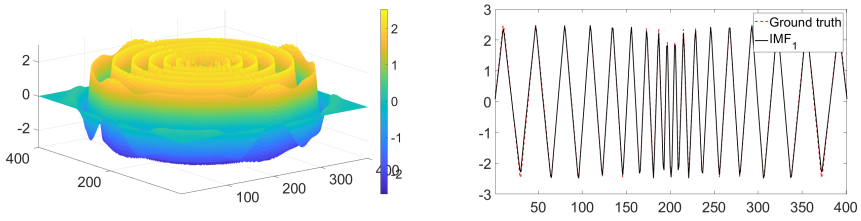
Figure 2.5 (a) is a 2D Artificial signal to be decomposed by the Multidimensional Iterative filtering algorithm. Figure 2.5 (b), is the cross section slice of the diagonal. The signal is composed by adding three functions.



(a) (b)

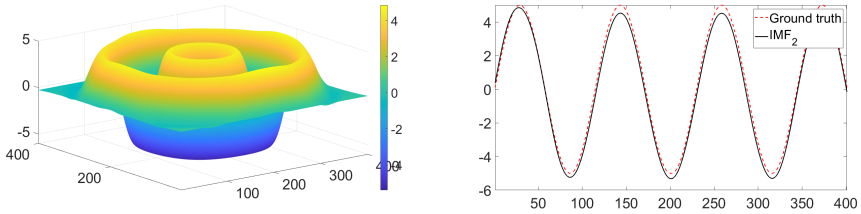
Figure 2.5: The Initial artificial Image

Figures 2.6 (a), 2.7 (a), and 2.8 (a) are decomposed plots of the 2D artificial signal. From the sections of the decomposed signal shown in figures 2.6 (b), 2.7 (b), and 2.8 (b), in black, we show the IMFs produced using the FIF2 algorithm. At the same time, the red curves represent the ground truth.



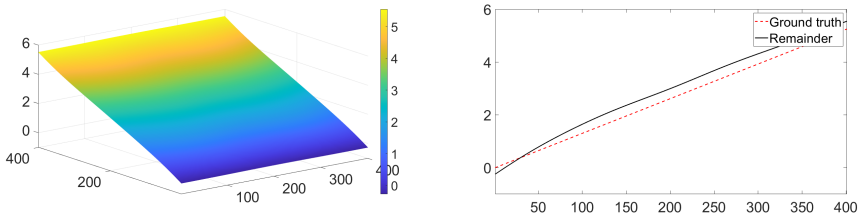
(a) (b)

Figure 2.6: IMF1



(a) (b)

Figure 2.7: IMF2



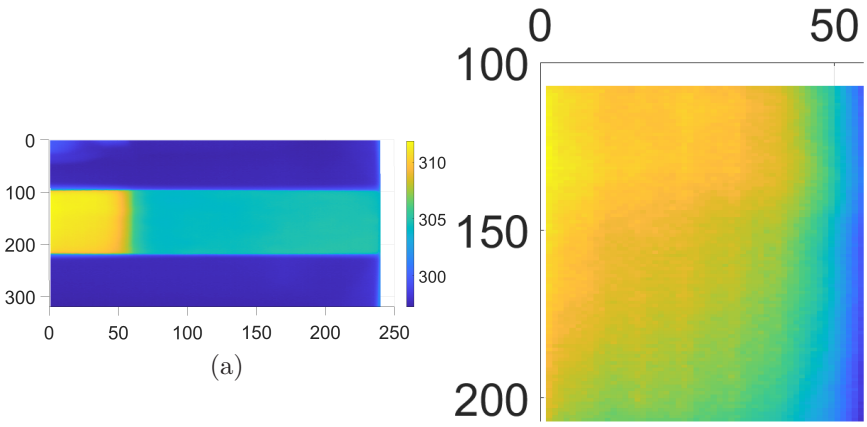
(a) (b)

Figure 2.8: Trend

Example 2

Real-life 2D thermal images of a composite material used for construction were recorded for 120 seconds for heating and 380 seconds for cooling. The images are to be subject to NDE on new composite material before being placed on the market [29]. This analysis is done to understand the reactions (i.e., development of defects) at the interface between two subsequent layers.

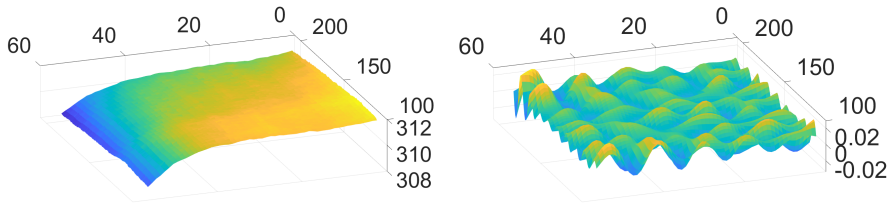
FIF2 is applied in the preprocessing stage to enhance thermal imprints, leading to improved subsurface features detection. FIF2 was applied due to its ability to produce robust results, which is fast compared to MEEMD [29]. There were 500 images collected, one per each second of the recording. Figure 2.9(a) shows the thermal image at the 66th second. Figure 2.9(b) shows the region of the thermal image to be preprocessed.



(a) (b)

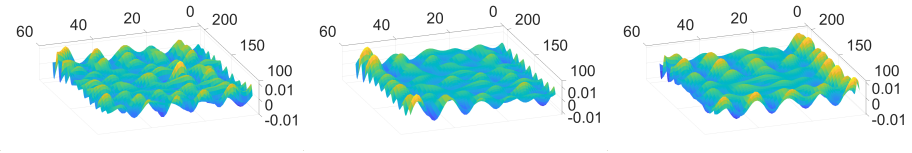
Figure 2.9: Real life 2D Signal

Figure 2.10(a) represents signal of the 66th thermal image. The boundary extension is applied asymmetrically, and then multidimensional FIF2 is used to produce 20 IMFs and a trend. Based on the study, IMFs 9, 10 and 11 were found to be significant for filtering as shown in Figure 2.11.



(a) (b)

Figure 2.10: Real life 2D Signal



(a) (b) (c)

Figure 2.11: IMFs of the real signal

Summing them gives the Figure 2.10(b) as the filtered signal. The obtained signal represents the grid hidden between two layers of the composite material. The identification of this grid is of particular importance in civil engineering applications [29].

CHAPTER 3

Machine Learning

In the age of computing, where computers do tasks instantaneously, computers need to have intelligence like humans. The human brain is a step ahead regarding imagination, inspiration, and common sense [38]. The sophisticated functionality of the brain, inspired the creation of artificial neural networks (ANN). The ANN tries to simulate the brain's neurons that process information in parallel. Recent application breakthroughs have been using ANN, including facial recognition system using PCA-ANN technique with feature fusion method [32], application to Language Translation using Colour Segmentation and Neural Network [2], application in the forecasting of time series [31].

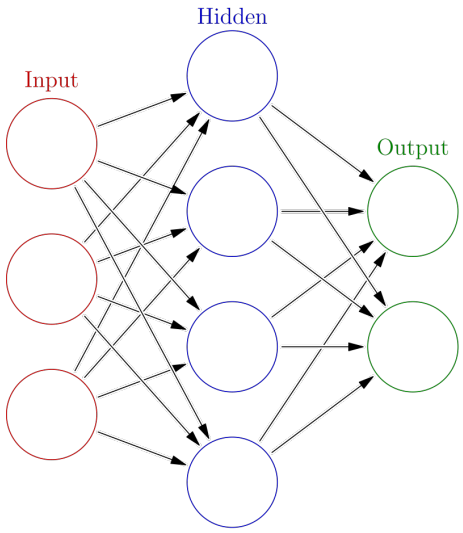
3.1 Artificial Neural Networks

Neural Networks was first developed in the 1950's [15] to try to make computers able to interpret real-world issues in a way the human brain does. ANN consists of an input layer of neurons, hidden layers of neurons(one or more), and the output layer of neurons. Figure 3.1 shows an artificial neuron network (ANN) where the lines connecting the neurons are associated with values called weights. The output, k_i, of neuron i in the hidden layer, is given by

$$k_i = \sigma\left(\sum_{i=1}^{N} V_{ij}x_j + T_i\right)$$

Where σ is the activation function, V_{ij} is the weights, x_j is the input to the input layer, N is the number of input neurons, and T_i is the threshold term for the hidden neuron i. The purpose of the activation function is to introduce nonlinearity and bound the neuron value, thus filtering out divergent neuron i. Several examples of activation functions include the tangent, sigmoid and hyperbolic tangent. The input is the independent variable, and the output is the dependent.

Figure 3.1: An example of Artificial Neural Network



Deep Learning is a class of Machine Learning techniques for learning in neural networks. These techniques allow the computers to learn from experience and understand through the hierarchy concept of acquiring knowledge. Deep learning can be classified as supervised, unsupervised, or semi-supervised; it depends on the aim of the neural networks [36].

Supervised learning predicts labelled output and input (the dataset is labelled), such as object detection and image classification [47]. In unsupervised learning, the dataset is unlabelled. Thus, the algorithm discovers autonomously anomalies and hidden patterns [23]. The semi-supervised learning has a large amount of unlabelled data with a small amount of labelled data with application, for instance, in analyzing lung sound [8].

3.2 Convolution Neural Networks

Introduced by Yann LeCun, in the 1980s, and used by many applications today from audio synthesis to image classification [7]. The advancement of computer vision has experienced exponential growth, primarily over the CNN algorithm. Convolution Neural Networks (CNNs), also known as ConvNet, are specialized neural networks that use convolution in place of general multiplication in at least one of the layers.

It involves taking input data and assigning importance (convolve) to various objects in the data and being able to differentiate them from one another. The data processed by CNNs have a grid-like structure such as 1D dimension time-series data, which is interpreted as 1D dimension grid taking sample steps at uniform time intervals. Also, the image data is a 2D dimension grid of pixels.

CNNs employ a mathematical operation called convolution. Consider a 1D convolution with function f as the input and function g as the kernel for 1D data such as time series. Then the output is given by

$$S(t) = (f * g) \cdot (t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau \tag{3.1}$$

Where the asterisk * represents the convolution for CNNs, g needs to be a valid probability density function for the output $S(t)$ to be a weighted average. In addition, g needs to be zero for all negative arguments to be compatible with real-world examples. In CNNs f is the input function, g is the kernel and $S(t)$ is the output also known as the feature map. The CNN equation is given by

$$S(t) = (f * g) \cdot (t) = \int_{0}^{t} f(\tau)g(t-\tau)d\tau \tag{3.2}$$

In practice, measurement at every instant time is not realistic. On the computer, time is discretized. Assuming t is discretized per second (or millisecond), the discrete convolution can be defined as:

$$(f * g) \cdot (t) = \sum_{0}^{n} f(\tau)g(t-\tau)d\tau \tag{3.3}$$

In deep learning, multidimensional data are prevalent and the kernel is a multidimensional array of parameters. Thus, convolution is used over more than one axis at a time. For example, a 2D image with I as our input and K as our 2D kernel, the convolution can be defined as:

$$\begin{aligned} S(i,j) = (I * K)(i,j) &= \sum_{m}\sum_{n} I(m,n)K(i-m, j-n) \\ &= \sum_{m}\sum_{n} K(m,n)I(i-n, j-m) \end{aligned} \tag{3.4}$$

By taking advantage of the commutative property of the convolution, we can write the convolution in two ways: the first equation works by sliding the image over the kernel, and the second equation works by sliding the kernel over the image. The result is a scalar value. We repeat the process for every point i, j for which the convolution can exist on the image. These values are stored in a convolved matrix.

CNN has three fundamental features to help improve the machine learning system: sparse iterations, parameter sharing, and equivariant representation [45].

Sparse Iteration

CNNs typically have sparse interactions between layers. In typical feedforward Neural Nets, every neuron in one layer is connected to every other in the next layer. Thus, a large number of parameters leads to many parameter estimations meaning a lot of training data will be needed. Convergence time increases due to increased computation, and the model may be overfitted.

ConvNtes reduce the number of parameters through indirect interactions. It is accomplished by making the kernel smaller than the input. Fewer parameters are stored, which reduces memory needed for the model, computing output requires fewer operations, and improves efficiency.

Parameter sharing

CNNs have parameters that are used for more than function in a model. Individual points in the same debt of the feature map (output) are created from the same kernel. It drastically reduces the number of learned parameters compared to typical Neural Networks. It does not affect the runtime for the forward operation, but it reduces the storage requirements for the model's parameters.

Equivariant Representation

A function f is equivariant with another function g if

$$f(g(x)) = g(f(x)) \tag{3.5}$$

Where x is the input, f and g can either be image translation or convolution operation. Then convolution is equivariant with respect to translation.

CNN layers consist of convolution, activation, pooling, and Fully connected layers (FCN).

Convolution Layer

It is where we convolve the image or data, in general, using filters as our kernel. Filters are small units we apply across the data through a sliding window. The depth of the data and filter are equal. Thus high-level features such as edges are extracted from the input. The convolution operation involves taking the element-wise product of filters and images and then summing those values for every sliding action to give a squashed convoluted feature output.

There are two types of results from the convolution operation. One, the dimensionality of the convolved feature is increased (or remains the same as the input), and second, the convolved feature's dimensionality is reduced—the former case results from applying the same padding, and the latter results from using valid padding.

Activation Layer

Only nonlinear activation functions are used between subsequent convolution layers because there won't be any learning if we use linear activation. This layer is also called the detector stage. Thus linear activation is passed through a nonlinear function such as the rectified linear activation function.

Consider A_1, A_2 be two convolution filters applied on X, input, without nonlinear activation between. Then

$$A_1 * (A_2 * x) = (A_2 * A_1) * x = A * x \tag{3.6}$$

Because of convolution's associative property of convolution, the two layers are as effective as one layer and just a single layer. This holds for typical Artificial Neural Networks. Typically reLU function is used for activation functions. ReLU is the abbreviation of rectified linear unit, the equation is given by

$$f(x) = x^{+} = max(0, x) \tag{3.7}$$

Where x is the neuron. It removes negative values by setting them to zero. Thus, non-linearity is introduced in the network. reLU is preferred over other functions because it gives significant penalty to generization accuracy and it is really fast because it does not activate all the neurons at the same time.

Pooling Layer

The pooling function modifies the output of the further layer by downsizing the sampling features to learn fewer parameters during training, where max pooling is the most common. It has two hyperparameters: the dimension of spatial extent and stride.

Dimension of spatial extent is the value of n such that we can take $n \times n$ feature representation and map it to a single value. The stride is how many features the sliding window skips along width and height. Pooling reduces the chances of overfitting as there are fewer parameters, memory size, and computation time. Also, pooling helps make the representation robust by contributing to local translation invariance.

Fully Connected Layer

The output from the convolution layers represents high-level features from the data. Adding FCN before the output is a cheap way of learning nonlinear combinations of these features. The output of a pooling layer: 3D feature map needs to be converted to input for the FCN: 1D feature vector. The 3D feature map is converted by flattening to a 1D vector.

In the classification, hidden layers are induced, and softmax activation is applied to the last layer of neurons, producing the voting weights for classification by predicting a multinomial distribution function.

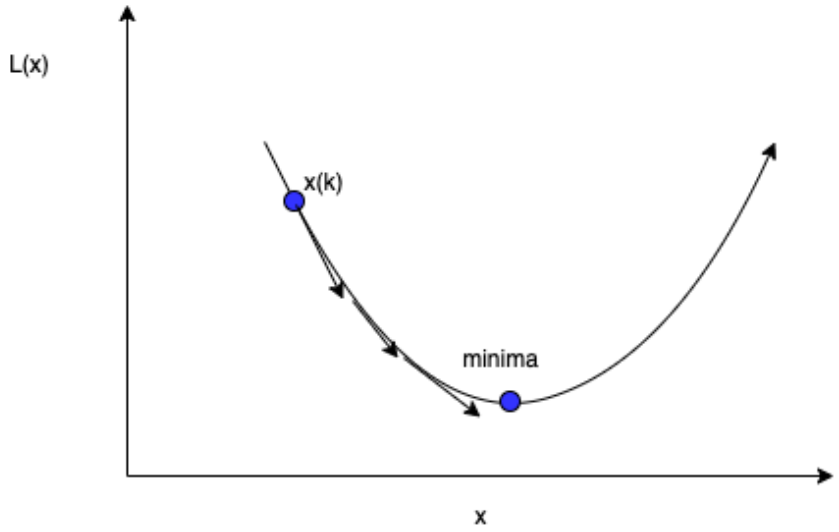
Back Propagation

The features in convolution layers and voting weights in fully connected layers come from back propagation. Deep Neural Networks does this on its own, no manual setting. The concept involves

$$Error = right\ answer - actual\ answer \tag{3.8}$$

The error is used to determine how much the network adjusts and changes. The learning process is how error helps drive the gradient descent. Gradient descent is a first-order iterative optimization algorithm for finding a local minimum of a differentiable error function. The amount of adjustment is determined by the size of the error. The adjustment is consistent for larger errors; the adjustment is limited for smaller errors, and no adjustment is required for zero error. The adjustment is made downhill the slope as shown in figure 3.2 where x is the weight and L(x) is the error.

Figure 3.2: Gradient Descent



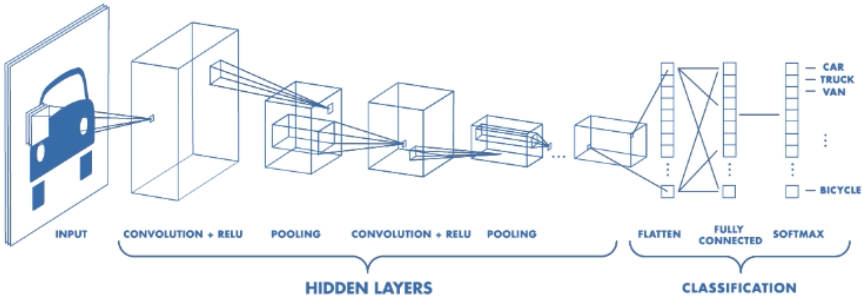
Doing this over many iterations helps all the weights settle to a minimum, the point where the network performance has maximum capacity. Any adjustment may induce the error to increase.

Hyperparameters

These are the knobs the network designer has to adjust manually. In the convolution we have size and number of features. In pooling we have window size and stride. In FCN we have number of neurons.

ConvNets are great in finding patterns and classifying images. Figure 3.3 [1] shows a typical CNN architecture.

Figure 3.3: Artificial Neural Network



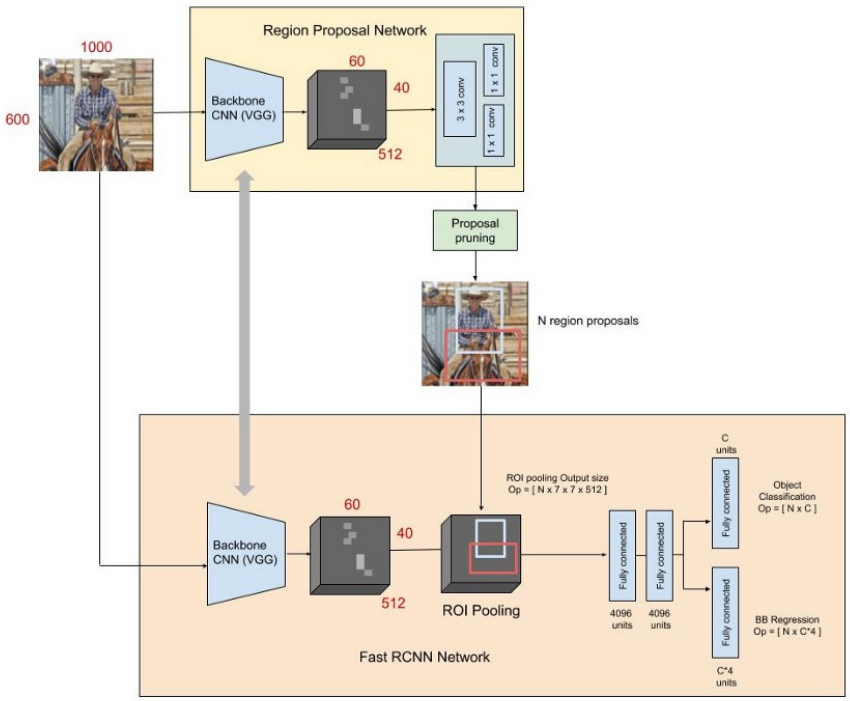
[1] Figure 3.3 source: https://uk.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html

3.3 Masked RCNN

Masked Region-based Convolution Neural Networks was introduced by Kaiming He, a researcher at Facebook AI in 2015 [21]. It works to solve the problem of instance segmentation in computer vision. Instance segmentation combines two subproblems: object detection and semantic segmentation. Object detection classifies individual objects and localizes each object instance using a bounding box. Semantic segmentation classifies each pixel into a fixed set of categories called a shaded mask without differentiating object instances. Since there are two phases for semantic segmentation the Masked RCNN has two parts: the object detection part that uses Faster-RCNN architecture and the semantic segmentation part that uses architecture similar to FCN. RCNN is an approach to bounding box object detection where Region of Interest (ROI) is created. An improved version, Faster RCNN incorporates attention mechanism using a region of proposed network (RPN) [27].

Object detection is performed in two stages. In Figure 3.4 [2], the bounding box is first determined using the RPN protocol to determine ROIs. Secondly, class labels are determined for each object by ROI pooling.

Figure 3.4: Faster RCNN Framework



Faster RCNN was not designed for pixel-to-pixel alignment; thus,when applying pooling on the ROI there is data loss. Pooling is used to downsample a feature and introduce invariant to minor input distortions such as an image's

[2]Figure 3.4 source:https://appliedsingularity.com/2021/06/08/object-detection-part-5-faster-r-cnn/

rotation. Stride is the number of steps we move during pooling. For a stride to be quantized, it means: Given an image of size 17×17, applying ROI pooling to the new size of 7×7. The stride s is given by

$$s = \frac{17}{7} = 2.42$$

The stride ROI pool is rounded off to 2. Thus 14×14 top part will be considered, the remaining part will be lost. Also, for a different size of images, such as 18×18. The stride s is given by

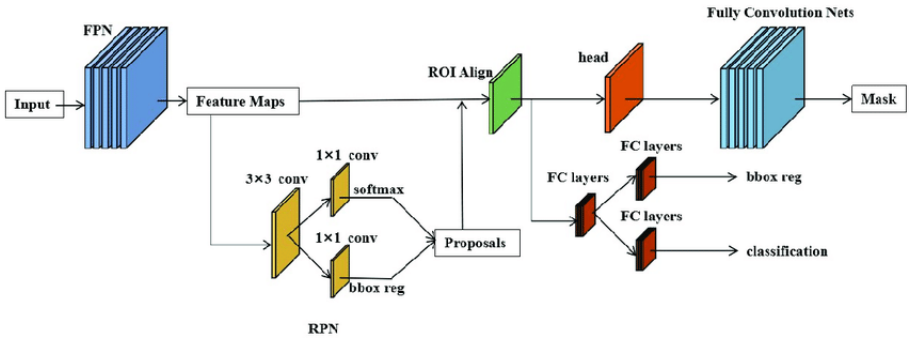
$$s = \frac{18}{7} = 2.57$$

Stride ROI pool is rounded off to 3, thus, leading to misalignment. To address this problem, a simple, quantized-free layer called ROI-align is used to preserve the spatial positions.

Thus Mask RCNN is similar to Faster RCNN; it additionally adds the output of the object mask using pixel-to-pixel alignment. In semantic segmentation, FCN is used to perform pixel-to-pixel multiclass categorization. It is used to produce the mask from each ROI convolution to maintain spartial orientation, which is crucial for a location-specific task.

In summary, instance segmentation involves object detection and semantic segmentation. Mask RCNN is a meta-algorithm that achieves instance segmentation by combining Faster RCNN and FCN. Mask RCNN has a special feature of using ROI-align to preserve features' spartial orientation with no data loss. This is summarized in Figure 3.5. [3].

Figure 3.5: Mask RCNN Framework



3.4 Transfer Learning

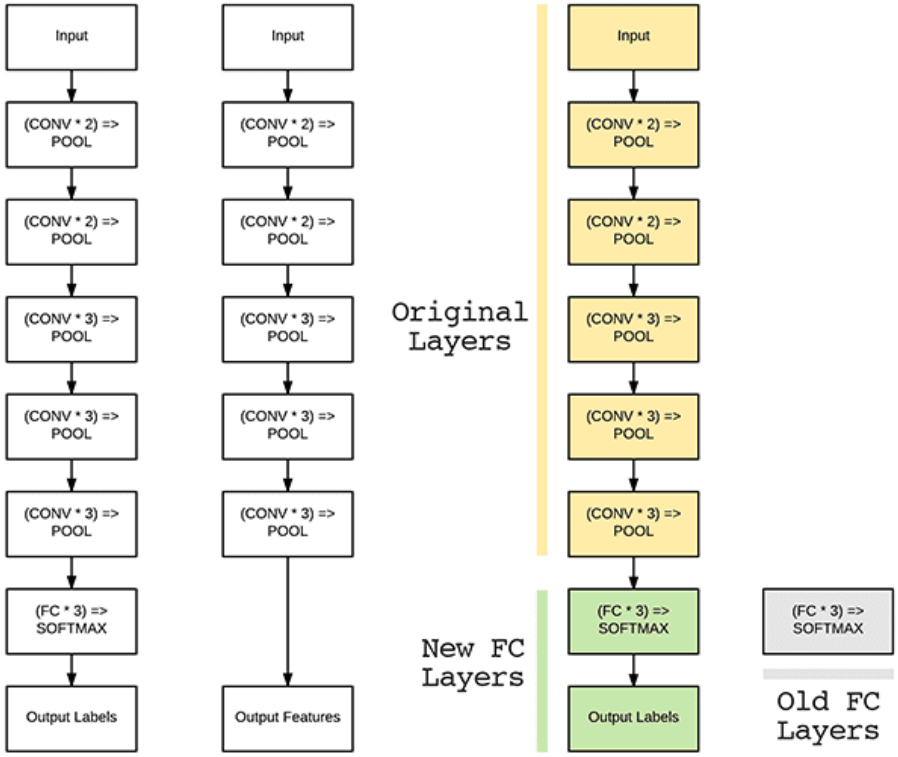
Transfer learning is a machine learning problem that focuses on storing knowledge gained while solving one problem and transferring it to a different but related problem [46]. It is prevalent in computer vision tasks. Researchers rarely train the deep learning model from scratch due to the limited amount of dataset. Meaningful parameters from the trained models with large datasets

[3]Figure 3.5 source: https://www.researchgate.net/figure/Mask-R-CNN-framework-for-instance-segment_fig2_334460507

are extracted in transfer learning. Transfer learning involves two scenarios: pre-training and fine-tuning.

The easiest way is to take the trained model, chop off its head layers and replace it with a random initialized one. Then train the parameter in the top layer of the task while other parameters remain fixed. The fixed parameters act as feature extractors and the top layer act as the typical FCN layer as shown in Figure 3.6 [4].

Figure 3.6: Transfer Learning pre-training architecture



Transfer learning works best if the data and task of the pre-trained model are similar to the data and task that needs training. In cases where there is not much data to train on the target task, transfer learning might be the only option to train a model without overfitting. Since having fewer parameters to train reduces the risk of overfitting [6].

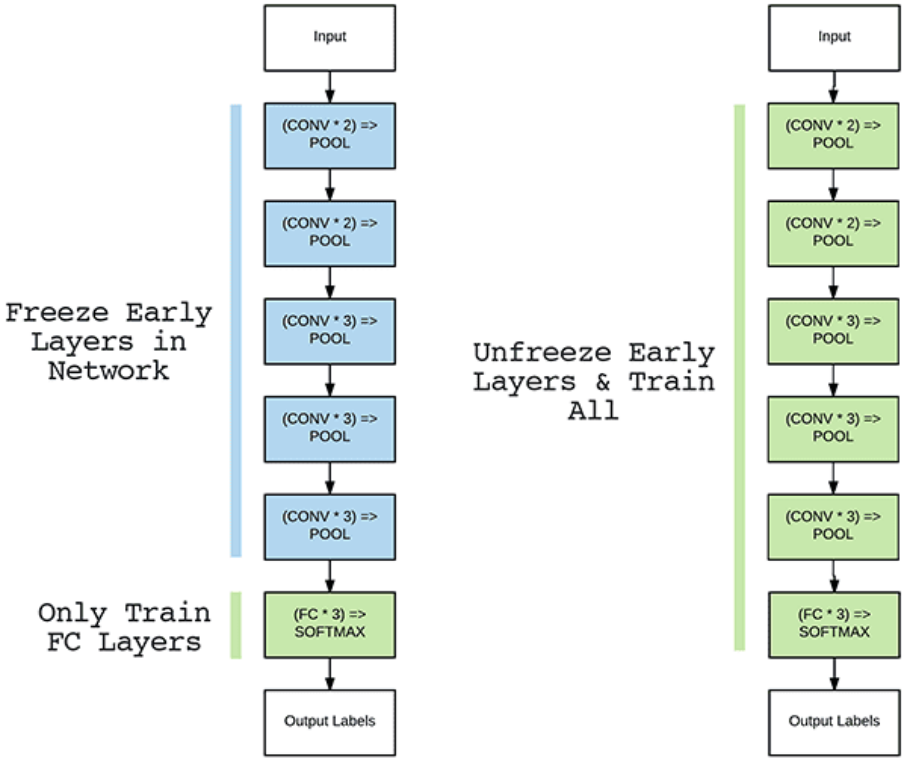
To improve performance and where there is more data, we can train the whole network by unfreezing the fixed parameters. It is known as fine-tuning. Here, the transfer is the initial value of the parameters. Initializing the weights from the pre-trained model instead of initializing them randomly can give the model an ahead start and speed up the convergence.

It is a common practice to start with frozen parameters and train only the randomly initialized head layers until they converge. Then unfreeze the fixed

[4] Figure 3.6 source: https://neptune.ai/blog/transfer-learning-guide-examples-for-images-and-text-in-keras

parameters and fine-tune the whole network as shown in Figure 3.7 [5]

Figure 3.7: Transfer Learning fine-tuning architecture



In transfer learning, if the data and task are not so close to the one of the pre-trained model, initializing the parameters using the pre-trained model is still better than random initialization.

The attractive property that makes transfer learning work is when deep learning networks are trained on large image data-sets, the early layer's parameters resemble each other regardless of the specific task the data-sets have been trained on [40].For example, the MS COCO is a large-scale image dataset containing 328,000 images of diverse objects. The data set was used to train the Mask RCNN model to recognize, label, detect and segment objects. Its weights are used in this project as initial parameter for transfer learning to train our model [41].

[5]Figure 3.7 source: https://neptune.ai/blog/transfer-learning-guide-examples-for-images-and-text-in-keras

PART II

The Second Part

CHAPTER 4

Numerical Applications

In this chapter, we demonstrate the performance of the FIF2 algorithm in preprocessing real-world $2D$ data. The preprocessed data were further analyzed through a deep learning technique called Mask RCNN in order to automatically detect defects on the surface. Due to energy conservation principle, the average temperature in the defect area is lower in the non-defect area since the defect is passive, i.e. not producing energy.

In the project, a non-destructive testing (NDT) approach called IRT was applied to detect material property without causing damage [34]. Since NDT does not permanently change and modify any sample being inspected, it saves on the cost and time needed for product inspection and research. Infrared thermography, a type of NDT, is widely used in defect detection for various materials.

4.1 Approaches in Infrared Thermography

Infrared thermography, also known as thermal imaging, is defined as imaging an object by sensing the infrared (thermal) radiation emitted by it. The technique maps temperature variations over the test object to detect both its surface, and subsurface defects [34].

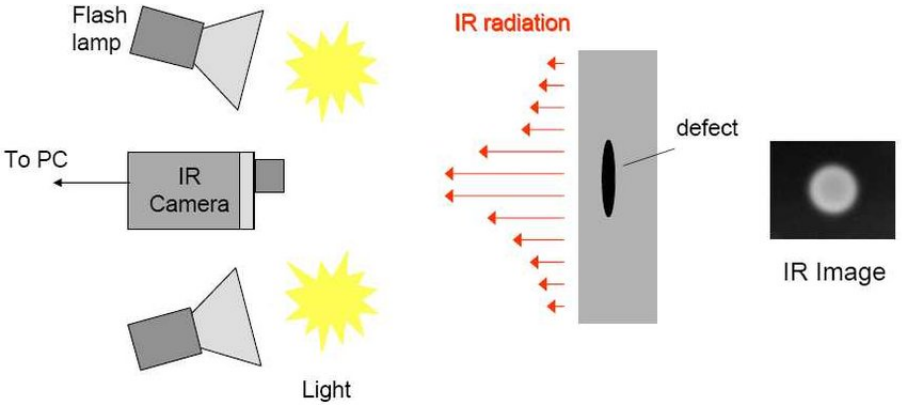
The temperature differentials on the surface or subsurface and the heat waves produced can be used to detect defects in the object's surface and subsurfaces.

Advantages of infrared are defects detected remotely from the acquisitions of images. This is made possible applying the thermal principle to which the object emits energy in the form of infrared radiation, which is not visible to the naked eyes when the temperature is above $\mathrm{T} = -273.15°C$, that is, absolute zero.

The temperature differentials, which can be detected without contact with the object, and the naked eye is not able to detect, are of the order of one-hundredth of a degree. This method requires the use of special cameras with detectors sensitive to infrared radiation and a lens that transmits infrared radiation. The infrared thermography camera lenses are made out of substances like germanium with low absorption in the infrared spectrum. It is necessary to use these specialized components instead of glass which is commonly used in photographic cameras because it is transparent in the infrared spectrum and has a high index of refraction. Glass is opaque in the infrared spectrum. Figure

4.1 [1] shows a typical infrared thermography system setup (active approach).

Figure 4.1: Active Thermography: Typical experimental setup



4.2 Infrared Thermography Classification

Infrared thermography can be classified as active or passive thermography. Active thermography measures the surface temperature of objects that require additional external thermal stimulation. Passive infrared thermography is a technique for inspecting objects surface and subsurface by monitoring their emitted thermal radiation without external stimulation. The objects are subject to natural heating by solar radiation.

In this project, passive infrared thermography was used for defect detection. The focus was on active defects which generate or absorb thermal energy, thus, can be detected in passive infrared thermography. The defective areas in the test object surrounding have different temperatures or abnormal hot spots as compared to the surrounding.

Performance of passive approach needs the following parameters to be taken into consideration such as the humidity of the surrounding, the temperature of the surrounding, deployment of a living organism, for example, presence of vegetation on the surface, condition of the surface, and so on [20].

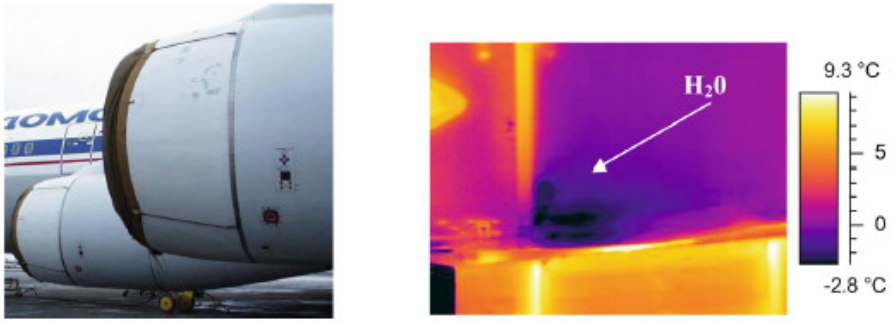
In passive thermography, the infrared camera supplies the thermal images. Each image has a value that corresponds to a particular temperature at a given time. The image is converted into false colors for better recognition of temperature contrasts. Usually, the blue color is utilized for the cold region and the red color for warm regions.

Passive infrared testing can be used, for example, to inspect areas that are difficult to access. Figure 4.2 [2] demonstrates that passive thermography was able to detect trapped water in composite cavities and also in aluminum honeycombs in the air inlet of an Ilyushin-96 airplane one hour after landing. [34].

[1] Figure 4.1 source: https://www.researchgate.net/figure/Typical-pulsed-thermography-system_fig4_229040003

[2] Figure 4.2 source: [34] page: 659

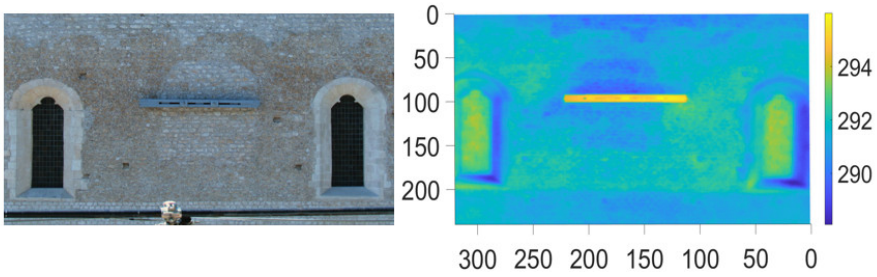
Figure 4.2: Passive IRT thermographic diagnostics of water ingress in the honeycomb aluminum air inlet of an Ilyushin-96 airplane



4.3 Data Acquisition

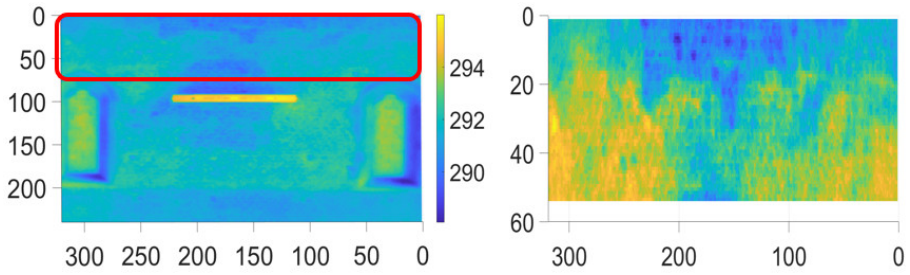
In this section, thermal images were collected from the Basilica di Santa Maria di Collemaggio wall. Basilica di Santa Maria di Collemaggio is a medieval church in the city of L'Aquila, central Italy, founded 1287 [28]. It was damaged during the earthquake of magnitude 6.2 Richter that stroke around central Italy on April 6, 2009. Recently there was a temporary repair on one part of the walls, as shown on the left image in Figure 4.3. Subsequently, the wall was completely repaired. Thermal images were recorded at a stationary location using the infrared camera at an interval of one minute per image for 500 minutes. The right image in Figure 4.3 is an example of thermal image of the wall at the 10th minute. The primary reason for data collection is to detect a defect on the subsurface, which may be a result of a detachment, water present in the wall, and so on.

Figure 4.3: Basilica di Santa Maria di Collemaggio (left side)



The region of interest (ROI) for the study was the upper part of the wall, excluding the windows and the horizontal metal bar, as shown in the left image of Figure 4.4. The metal bar was primarily excluded because it produces high amplitude and frequency waves that are not suitable for the FIF2 algorithm producing false results. The right image in Figure 4.4 is the ROI of the thermal image at the 10th minute which we want to preprocess. The ROI for each thermal image was recorded.

Figure 4.4: Thermal image recorded at the 10th minute



4.4 FIF2 application to Data preprocessing

In this thesis, FIF2 was deployed in preprocessing thermal images to minimize the negative impact of temperature variations at the borders and the high-frequency oscillations due to noise made by the camera, thus, providing clear and precise thermal imprints to be segmented. FIF2 was used due to its fast and unique algorithm that produced reliable results as shown in [29].

FIF2 was applied, with parameters being set at default and some manually adjusted. Delta, which is FIF2 stopping criterion, was set at 0.0001. "ExTPoints" which is the number of extrema allowed at the remainder, was set at 3. "ExtensionType", which is the type of boundary extension, was set at "asym" meaning antisymmetric padding (half point). "Xi", which allows to fine tune mask length was set at 1.6. "Alpha" used for the mask length computation was set to "Almost min" which is a value close to 30-th percentile.

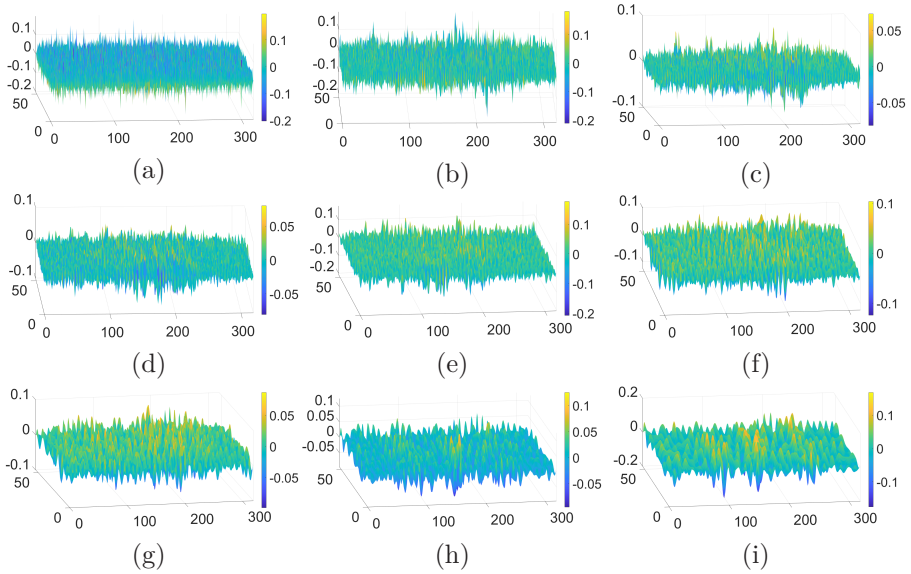
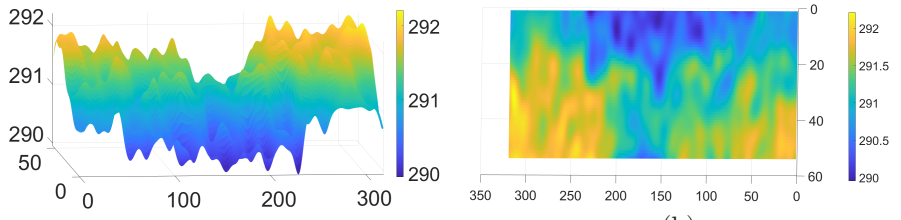


Figure 4.5: IMFs of the thermal image minute 10th. Panels (a) to (i) show IMFs from 1 to 9, respectively.

It took 20 min to decompose all 500 images on a Windows computer with

Matlab version 2020b, and an Intel processor core $i5$ version 3. Figure 4.5 shows the IMFs produced except the trend line, where (a)-(i) represents 1st-9th IMFs respectively. The IMFs had high frequency and small amplitudes, which represented the noise.

The last IMF is considered as the clean Thermal image. Figure 4.6(a) shows the 3D view of the last IMF. Figure 4.6(b) shows the 2D clean thermal image at the 10th minute.



(a) (b)

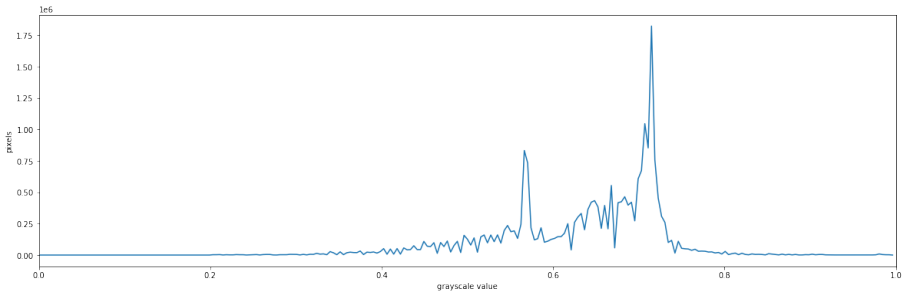
Figure 4.6: The preprocessed thermal image of the $10th$ min

4.5 Binary Thresholding

There are many techniques for analyzing defects in thermal images. Among them, binary thresholding is the simplest and easiest method, which binarizes an image using a threshold value. It is used in many pre-processing steps in image processing, such as extracting only pixels with a brightness value above a certain level, separating the background from objects in an image, or simplifying all the information in an image. In this thesis, automatic binary thresholding using the Otsu algorithm is applied.

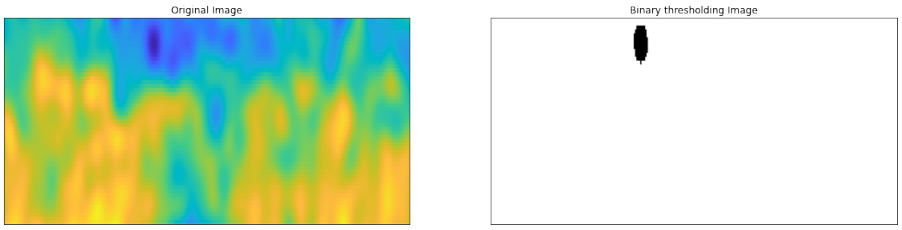
The Otsu algorithm automatically calculates the optimal threshold for classifying images into two classes using a histogram based on a grayscale. To classify the thermal image into two categories, an appropriate threshold value, k, is applied such that pixels with grayscale values on one side of k will be turned "on". In contrast, pixels with grayscale values on the other side will be turned "off". It is beneficial for situations where the grayscale histogram of an image has two peaks that correspond to the background and objects of interest. Figure 4.7 shows the grayscale histogram of the thermal images.

Figure 4.7: Grayscale Histogram



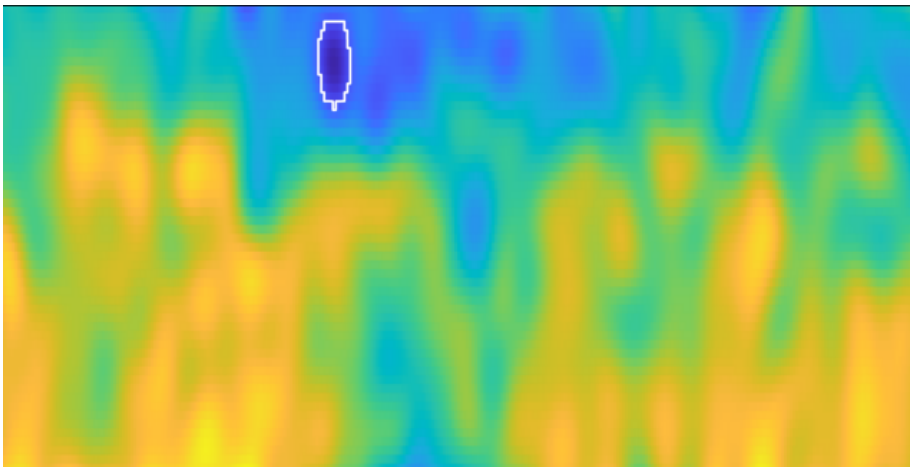
The histogram has small peaks starting around 0.3 and two significant peaks, one between 0.5 and 0.6, the other peak very near 0.7. Thus, this image is a good candidate for thresholding with Otsu's method. The outcome is that Otsu's method finds a threshold value as k = 0.45 of the grayscale histogram. Now we can create a binary mask with the comparison operator ">". Pixels above the threshold value are turned on, and those below the threshold are turned off. Finally, we use the mask to select the foreground as shown in Figure 4.8.

Figure 4.8: Binary thresholding applied to thermal image recorded at 401th minute



Contours are used to get and plot the external boundary found with binary thresholding. A typical application of contour is to detect geometrical shapes in images, which can help simplify problems involving classification or object detection. Contours are rarely simple polygons depending on the quality, angle, noise, and many other factors. Contours try to simplify the shape of the defect rather than precisely follow the defect shape. The binary images we got from the binary thresholding are used to find the contours of the defects. Next, contours were drawn on the original thermal images as shown in Figure 4.9.

Figure 4.9: Application of Contours



4.6 Mask RCNN application to Data postprocessing

In this section, we will highlight the application Mask RCNN in automatic defect detection and classification. Python was our choice programing language, and

because Mask RCNN is a technique for detection using a convolutional neural network, we used the python Keras library and the TensorFlow framework used for machine learning and artificial intelligence.

A dataset is a collection of data, which we divided into training, validation, and testing datasets to automatically detect defects in a ratio of $8:1:1$. The images were standardized to be the same size of 1657×907 pixels, while a training algorithm was used to detect the defects and classify them as either considerable or minor. A minor defect means it does not reduce the usability or stability of the wall and, therefore, can be ignored. Whereas, a considerable defect means it may affect the stability or usability of the wall in the long run and thus may need more check-ups.

Mask RCNN is an instance segmentation method extended from Faster RCNN, which has a convolution backbone architecture. ROI Align preserves the spatial orientation of features with no data loss for extraction over the entire network image. This approach efficiently detects objects in an image while generating a high-quality segmentation mask for each instance. Due to predefined anchors, the speed of the model is slow.

Training

Training refers to teaching an algorithm toward a specific task for which it will be applied. These algorithms also learn from experience without being explicitly programmed. In our experiment, we labeled 500 images using the VGG annotator, each with a varying number, classification, and size of defects [16]. We trained our algorithm using four hundred images.

The training processing was conducted on a GeForce GTX1080TI, and it took about 8 hours to train the network heads and then about 6 hours to fine-tune the whole model by introducing all the network layers. The operating system is set as windows 10. The framework of the learning model is set as COCO-Model. CPU: i5-9018k. Memory: 16GB, GPU: NVIDIA GeForce GTX1080TI. Figure 4.10 shows the summary of the training model.

Figure 4.10: Summary of the detection model developed in this work

```
Starting at epoch 0. LR=0.001

Checkpoint Path: ./defect_cfg20220720T0132/mask_rcnn_defect_cfg_{epoch:04d}.h5
Selecting layers to train
fpn_c5p5               (Conv2D)
fpn_c4p4               (Conv2D)
fpn_c3p3               (Conv2D)
fpn_c2p2               (Conv2D)
fpn_p5                 (Conv2D)
fpn_p2                 (Conv2D)
fpn_p3                 (Conv2D)
fpn_p4                 (Conv2D)
In model:  rpn_model
    rpn_conv_shared        (Conv2D)
    rpn_class_raw          (Conv2D)
    rpn_bbox_pred          (Conv2D)
mrcnn_mask_conv1       (TimeDistributed)
mrcnn_mask_bn1         (TimeDistributed)
mrcnn_mask_conv2       (TimeDistributed)
mrcnn_mask_bn2         (TimeDistributed)
mrcnn_class_conv1      (TimeDistributed)
mrcnn_class_bn1        (TimeDistributed)
mrcnn_mask_conv3       (TimeDistributed)
mrcnn_mask_bn3         (TimeDistributed)
mrcnn_class_conv2      (TimeDistributed)
mrcnn_class_bn2        (TimeDistributed)
mrcnn_mask_conv4       (TimeDistributed)
mrcnn_mask_bn4         (TimeDistributed)
mrcnn_bbox_fc          (TimeDistributed)
mrcnn_mask_deconv      (TimeDistributed)
mrcnn_class_logits     (TimeDistributed)
mrcnn_mask             (TimeDistributed)
```

The transfer learning technique was applied first on the heads using weights from the pre-trained COCO dataset model. The training detail of the Mask-RCNN was set as follows: The weight decay is set to 0.0001, Network training used Resnet101 as the backbone, The mini mask size is 512x512, and the loss weight is equal for each class and mask (RPN class, RPN bounding box, MRCNN class, MRCNN bounding box, and MRCNN mask), The learning momentum is 0.9 and learning rate is 0.0002.

The training of the first five epochs of network heads was followed by the training of all network layers for six epochs. Then training was done on the whole network for six epochs to increase the robustness and accuracy of the algorithm. The learning rate was modified by reducing it by a 10th, to avoid overfitting.

4.7 Evaluation metric

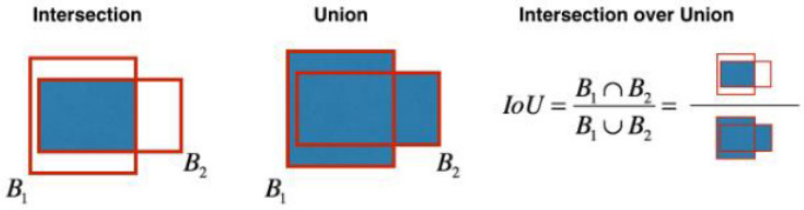
Intersection over Union (IOU)

IOU is an evaluation metric (between 0 and 1) that measures the overlap between the boundaries of the ground truth of annotation and the predicted edge. IOU evaluates whether a prediction is “good enough” [39]. A prediction is considered True Positive if IOU $>$ threshold and False Positive if IOU $<$

threshold. The closer the prediction is to 1, the closer it is to perfection. Figure 4.11 illustrates the graphical view of the equation below.

$$\text{IOU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{4.1}$$

Figure 4.11: Graphical View of the IoU equation



Mean Average Precision (mAP)

To understand mAP, we review precision and recall first. The recall is the True Positive Rate; that is, of all the actual positives, how many are True positives predictions. In the problem under investigation it means how the system correctly recognized the defects over the cases that actually contained them. Precision is the Positive prediction value; that is, of all the positive predictions, how many are True positives predictions. In the case of defect identification, precision means the ratio from the cases that contain the defects over the cases that are recognized by the system that includes the defects, which represents how accurate the system is in identifying the defects. F1 Score finds the most optimal confidence score threshold, where precision and recall give the highest F1 Score. The F1 Score calculates the balance between precision and recall. If the F1 Score is high, precision and recall are high, and vice versa.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} = \frac{\text{True Positive}}{\text{number of ground truths}} \tag{4.2}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} = \frac{\text{True Positive}}{\text{number of predictions}} \tag{4.3}$$

$$\text{F1 Score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \tag{4.4}$$

The precision and recall values are then plotted to get a PR (precision-recall) curve. The area under the PR curve is called the Average Precision (AP). The PR curve follows a kind of zig-zag pattern as recall increases absolutely, while precision decreases overall with sporadic rises. The mAP is the averaged AP over all the object categories. Thus it is used as a standard metric to analyze the accuracy of an object detection model.

4.8 Results

This model provided the shape and location of each defect detection result based on the labeled images containing the ground truth. During each epoch, the confidence metric improved over time, making detection closer to perfect.

The Mask RCNN showed some indistinguishable results from the ground truth when the number of defects was less than three per image. Several shapes and classes of defects are detected. Since the training database is composed of irregular shapes, the testing results for thermal images with three defects are not detected accurately. It indicates that the mask RCNN detection model does not enhance detection performance based on instance segmentation of pixel-to-pixel alignment.

The algorithm precision, Recall, and F-score analysis results are shown below.

Precision = 0.71

Recall = 067

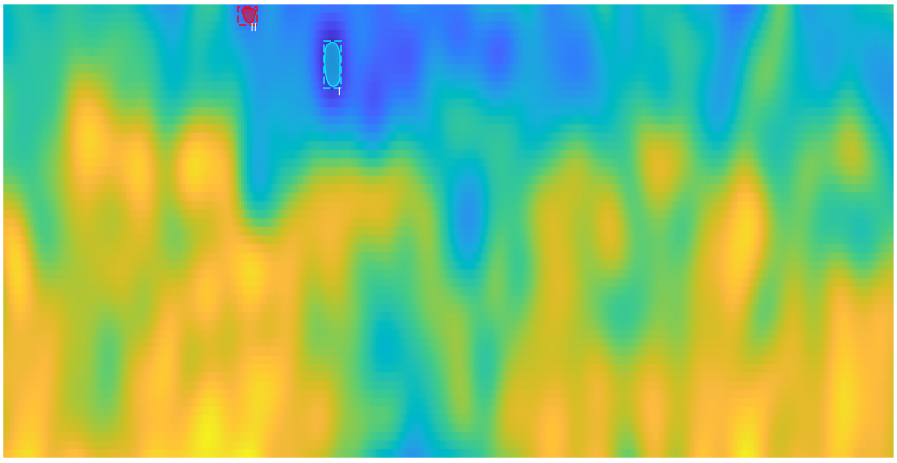
F1 Score = 0.69

mAP = 0.72

The confidence score threshold was set to 85% of the probability of detection to distinguish True Positive, False Positive, True Negative, and False Negative cases. This study aims to extract automatically, segment the features and classify (i.e., defects) that could be found in each thermal frame. Each classification may either be a major defect or a minor defect region in this task.

Figure 4.12 shows the classification of defects using the mask RCNN algorithm. Mask (I), light blue in color, is the major defect with darkest blue region indicating inspection is needed. Mask (II), red in color, is the minor defect with dark blue but less darker than in mask(I) indicating immediate defect inspection might not be warranted. They were detected at 91.57% and 90.89%, respectively. The presence of noise is the main factor affecting the segmentation results. For checking (i.e., a double-check), the inspection may be performed by a professional restorer or by using an additional NDT technique (in contact or telemetric). The segmentation boundary still does not capture the full defect size and shape. But it showed performance improvement in defect detection capability on defects with regular shape, small size, and more damage.

Figure 4.12: Segmentation result from instance segmented Mask RCNN algorithm



Therefore, it indicates that the detection performance for shape and classification using mask RCNN has significantly improved.

4.9 Discussion and result analysis

To implement a robust detection model and effectively improve its accuracy, one has to increase the size of the dataset. Also, the Data augmentation technique, which involves rotation, horizon flipping, and vertical shifts, improves the robustness of the model. It would also enhance the robustness of the algorithm to train for the detection of large-size defects and improve the results of thermal images.

Mask RCNN involves a pixel-based marking approach that could mark the defects accurately, as opposed to marking a large area around each defect as in binary thresholding. But due to the nature of thermal images being RGB format, it misses creating an exact shape of the defect in the bounding box.

Therefore, building and creating more diverse and representative training samples is crucial in future work in this research. A good defect characterization is essential to avoid replacing parts that could be left and to avoid leaving critically damaged components without the needed repair. Therefore, these results are essential, especially in the designing of autonomous diagnosis NDT systems, which can make decisions by themselves regarding the integrity of the inspected part.

Future work in this research direction:

(i) improving by tuning the network parameters of the (Mask RCNN) for instance segmentation.

(ii) since the CNN technique achieves excellent performance, other network architectures should be tested and compared in the future to identify the best tool for defect measurement with infrared images.

CHAPTER 5

Conclusion and Recommendation

5.1 Overview

In this work a pre-processing technique based on FIF and deep learning algorithm mask RCNN ware applied for automatic defect detection in infrared thermography. The FIF decomposition method handles nonstationary signals better than Fourier and wavelet transform. Chapter 2 introduces a short theory review, and examples of application of the FIF algorithm. Based on the results and analysis reported in Chapter 4, FIF performs well in cleaning the thermal images. Thus, it improves defect detection. Due to the need for defect detection in infrared thermography, deep learning tools are used. Mask RCNN algorithm was applied due to its ability to Instance segment the defects. Mask RCNN was introduced in Chapter 3 and its results are reported in Chapter 4. Overall, the proposed approaches appear to be promising to improve defect detection and automation of the process.

5.2 Limitations of Work

The FIF decomposition method also has a drawback: the "end effects" at the boundaries. This drawback affects any signal decomposition methods which deal with compactly supported signals resulting in the loss of information at and nearby the edges of the dataset during the decomposition of signals. To increase the decomposition method accuracy, extending signals across the boundaries is essential. However, currently there are just a limited number of extension approaches available in the literature. The generalization of the extension techniques may lead to a significant improvement in reducing the boundary effects. The manual selection of the IMFs to be chosen from the FIF decompositon is another limitation. Automating this process and other parameters selection will save time and optimize the algorithm performance.

In this work, the Mask RCNN algorithm was applied to the older Tensorflow framework making the process slow. It becames difficult in this way to use multiple modifications, such as increasing layers and modifying activation functions.

5.3 Future Work and Recommendation

Future work will focus on automating the FIF algorithm and to improve and fasten the pre-processing of the thermal images. In addition, automatic defect detection of more complicated structured materials through the optimization and combination of different deep learning models should be studied. Deep learning tools will be applied using up-to-date frameworks, and more data will be collected and analyzed to improve the proposed models.

Bibliography

[1] Abdelouahad, A. et al. 'Reduced reference image quality assessment based on statistics in empirical mode decomposition domain'. In: (2020), p. 15.

[2] Akmeliawati, R., Ooi, M. P.-L. and Kuang, Y. C. 'Real-Time Malaysian Sign Language Translation using Colour Segmentation and Neural Network'. In: *2007 IEEE Instrumentation Measurement Technology Conference IMTC 2007*. 2007, pp. 1–6.

[3] Amit, S., Yoel, S. and Boaz, N. In: *SIAM Journal on Imaging Sciences* vol. 2, no. 1 (2009). Publisher Copyright: © 2009 Society for Industrial and Applied Mathematics., pp. 118–139.

[4] Atiyah, M. F. and Macdonald, I. G. *Introduction to commutative algebra*. Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont., 1969, pp. ix+128.

[5] Avdelidis, N. and Moropoulou, A. 'Applications of infrared thermography for the investigation of historic structures'. In: *Journal of Cultural Heritage* vol. 5, no. 1 (2004), pp. 119–127.

[6] Bengio, Y. 'Deep learning of representations for unsupervised and transfer learning'. In: *Proceedings of ICML workshop on unsupervised and transfer learning*. JMLR Workshop and Conference Proceedings. 2012, pp. 17–36.

[7] Bruna, J. et al. 'Spectral networks and locally connected networks on graphs'. In: *arXiv preprint arXiv:1312.6203* (2013).

[8] Chamberlain, D. et al. 'Application of semi-supervised deep learning to lung sound analysis'. In: *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2016, pp. 804–807.

[9] Cicone, A. 'Iterative filtering as a direct method for the decomposition of nonstationary signals'. In: *Numerical Algorithms* vol. 85, no. 3 (2020), pp. 21–27.

[10] Cicone, A., Li, W. and Zhou, H. 'New theoretical insights in the decomposition and time-frequency representation of nonstationary signals: the IMFogram algorithm.' In: *Numerical Analysis* vol. v4, no. 15 (2022), p. 23.

[11] Cicone, A., Liu, J. and Zhou, H. 'Adaptive Local Iterative Filtering for Signal Decomposition and Instantaneous Frequency analysis'. In: *Mathematical Numerical Analysis* vol. 31, no. 92 (2015), p. 31.

[12] Cicone, A., Liu, J. and Zhou, H. 'Hyperspectral chemical plume detection algorithms based on multidimensional iterative filtering decomposition'. In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* vol. 374, no. 2065 (2016), p. 20150196.

[13] Cicone, A. and Zhou, H. 'Multidimensional Iterative Filtering Method for the Decomposition of High–Dimensional Non–Stationary Signals'. In: *Journal of Numerical Mathematics: Theory, Methods and Applications* vol. 10, No. 2 (2017), pp. 278–298.

[14] Cicone, A. and Zhou, H. *Numerical Analysis for Iterative Filtering with New Efficient Implementations Based on FFT*. 2018.

[15] Dreyfus, S. E. 'Artificial neural networks, back propagation, and the Kelley-Bryson gradient procedure'. In: *Journal of Guidance, Control, and Dynamics* vol. 13, no. 5 (1990), pp. 926–928. eprint: https://doi.org/10.2514/3.25422.

[16] Dutta, A. and Zisserman, A. 'The VGG Image Annotator (VIA)'. In: (Apr. 2019).

[17] Fang, Q. et al. 'Automatic defect detection in infrared thermography by deep learning algorithm'. In: *Thermosense: thermal infrared applications XLII*. Vol. 11409. SPIE. 2020, pp. 180–195.

[18] Flandrin, P. *Time-frequency analysis*. Academic press, 1998.

[19] Ghosh, K. K. and Karbhari, V. M. 'A critical review of infrared thermography as a method for non-destructive evaluation of FRP rehabilitated structures'. In: *International journal of materials and product technology* vol. 25, no. 4 (2006), pp. 241–266.

[20] Harizi, W. et al. 'Mechanical damage assessment of Glass Fiber-Reinforced Polymer composites using passive infrared thermography'. In: *Composites Part B: Engineering* vol. 59 (2014), pp. 74–79.

[21] He, K. et al. 'Mask R-CNN'. In: *CoRR* vol. abs/1703.06870 (2017). arXiv: 1703.06870.

[22] Janssens, O. et al. 'Deep Learning for Infrared Thermal Image Based Machine Health Monitoring'. In: *IEEE/ASME Transactions on Mechatronics* vol. 23 (July 2017), pp. 151–159.

[23] Karhunen, J., Raiko, T. and Cho, K. 'Unsupervised deep learning: A short review'. In: *Advances in independent component analysis and learning machines* (2015), pp. 125–142.

[24] Kunal, P. and Kunal, M. '"Medical Signal Processing in Biomedical and Clinical Applications"'. In: *Journal of Healthcare Engineering* vol. vol. 2018, no. 920 (2018), p. 2.

[25] Lv, Y. and Zhang, Y. 'Optimized Adaptive Local Iterative Filtering Algorithm Based on Permutation Entropy for Rolling Bearing Fault Diagnosis'. In: *Mathematical Numerical* vol. 20, no. 920 (2018), pp. 17–23.

[26] Mehl, R. W. 'Applications of Advanced Signal Processing Techniques in Pure and Applied Geophysics'. In: *University of Connecticut Graduate School at OpenCommons@UConn* vol. -, no. 939 (2018), p. 92.

[27] Ren, S. et al. 'Faster r-cnn: Towards real-time object detection with region proposal networks'. In: *Advances in neural information processing systems* vol. 28 (2015).

[28] Sabatino et al. 'The Basilica of Santa Maria di Collemaggio in LAquila Modern Provisional Measures'. In: *Journal of civil engineering and architecture* vol. 8 (2014), pp. 56–65.

[29] Sfarra, S. et al. 'Maximizing the detection of thermal imprints in civil engineering composites via numerical and thermographic results pre-processed by a groundbreaking mathematical approach'. In: *International Journal of Thermal Sciences* vol. 182 (2021), p. 23.

[30] Stallone, A., Cicone, A. and Materassi, M. 'New insights and best practices for the successful use of Empirical Mode Decomposition, Iterative Filtering and derived algorithms'. In: *Scientific reports* vol. 10, no. 1 (2020), pp. 1–15.

[31] Tealab, A. 'Time series forecasting using artificial neural networks methodologies: A systematic review'. In: *Future Computing and Informatics Journal* vol. 3, no. 2 (2018), pp. 334–340.

[32] Toufiq, R. and Islam, M. R. 'Face recognition system using PCA-ANN technique with feature fusion method'. In: *2014 International Conference on Electrical Engineering and Information Communication Technology*. 2014, pp. 1–5.

[33] Usamentiaga, R. et al. 'Infrared Thermography for Temperature Measurement and Non-Destructive Testing'. In: *Sensors (Basel, Switzerland)* vol. 14 (July 2014), pp. 12305–12348.

[34] Vavilov, V. and Burleigh, D. *Infrared thermography and thermal nondestructive testing*. Springer, 2020.

[35] Wang, G. et al. 'On Intrinsic Mode Function.' In: *Advances in Adaptive Data Analysis* vol. 2 (July 2010), pp. 277–293.

[36] Wang, H. and Raj, B. 'On the origin of deep learning'. In: *arXiv preprint arXiv:1702.07800* (2017).

[37] Wang, M. and Jinquiao, D. 'Smooth solution of a nonlocal Fokker–Planck equation associated with stochastic systems with Lévy noise'. In: *Applied Mathematics Letters* vol. 58 (2016), pp. 172–177.

[38] Wang, S.-C. 'Artificial Neural Network'. In: *Interdisciplinary Computing in Java Programming*. Boston, MA: Springer US, 2003, pp. 81–100.

[39] Wang, X. et al. 'Detection and localization of image forgeries using improved mask regional convolutional neural network'. In: *Mathematical Biosciences and Engineering* vol. 16, no. 5 (2019), pp. 4581–4593.

[40] Weiss, K., Khoshgoftaar, T. M. and Wang, D. 'A survey of transfer learning'. In: *Journal of Big data* vol. 3, no. 1 (2016), pp. 1–40.

[41] Wu, H., Gao, W. and Xu, X. 'Solder joint recognition using mask R-CNN method'. In: *IEEE Transactions on Components, Packaging and Manufacturing Technology* vol. 10, no. 3 (2019), pp. 525–530.

[42] Wu, Z., Huang, N. E. and Chen, X. 'The multi-dimensional ensemble empirical mode decomposition method'. In: *Advances in Adaptive Data Analysis* vol. 1, no. 03 (2009), pp. 339–372.

[43] Xia, Y. and Zhang, B. 'Bidimensional Multivariate Empirical Mode Decomposition With Applications in Multi-Scale Image Fusion'. In: *Numerical Algorithms, IEEE* vol. 10, no. 1109 (2019), pp. 10–13.

[44] Xu, Y. et al. 'Automatic defect detection and segmentation of tunnel surface using modified Mask R-CNN'. In: *Measurement* vol. 178 (2021), p. 109316.

[45] Zhang, M. et al. 'The application of one-class classifier based on CNN in image defect detection'. In: *Procedia computer science* vol. 114 (2017), pp. 341–348.

[46] Zhao, W. 'Research on the deep learning of the small sample data based on transfer learning'. In: *AIP Conference Proceedings*. Vol. 1864. 1. AIP Publishing LLC. 2017, p. 020018.

[47] Zhao, Z.-Q. et al. 'Object detection with deep learning: A review'. In: *IEEE transactions on neural networks and learning systems* vol. 30, no. 11 (2019), pp. 3212–3232.