Sensor fusion based condition monitoring of induction motor using artificial intelligence techniques

Sarvavignoban Sandirasegaram

School of Electrical Engineering

Thesis submitted for examination for the degree of Master of Science in Technology.
Espoo 7.5.2020

Supervisor

Prof. Anouar Belahcen

Advisor

Alireza Nemat Saberi, M.Sc.
Industries are proliferating, and the need for induction motors (IMs) plays an essential role in various technical and economic fields. Due to the excessive usage, several faults occur in the machine. It is imperative to detect faults at an early stage. Condition monitoring techniques can detect unexpected failures in the machine. Otherwise, they may cause massive losses for industries. Several methods have been investigated using a single sensor and signal-processing methods. However, using a single sensor does not facilitate in detecting different types of faults. Therefore, several current and vibrations sensors are used to identify various forms of defects. However, data derived from this approach is too vast and complicated. A number of artificial intelligence technologies are being used to solve this. The support vector machine (SVM) has been used for this thesis due to its excellent classification efficiency compared to other techniques.

The main focus of the thesis is on the development of a sensor fusion-based approach for condition monitoring of IMs using SVM. This was achieved by adequately preprocessing the dataset, extracting the maximum number of features, choosing the suitable features, identifying the hyperparameters and designing the model. The thesis investigates three main processes: binary classification, multiclass classification, and sensor fusion. The algorithm’s working condition has been checked in binary classification, and feature selection methods have been evaluated. The accuracy of different classes is monitored using a single sensor in a multiclass classification method. The sensor fusion approach has been used to test accuracy changes with the fusion process at the feature level. The fused multi-dimensional information is used to train a multiclass SVM with eight classes. Finally, the proposed approach is tested by current and vibration data. The results prove that the feature level fusion method has great potential in increasing the accuracy of fault diagnosis.
Preface

I take this opportunity to thank my Professor Anouar Belahcen, for giving me the opportunity and guidance to work on this thesis. It was such an honour for me to know you, and be part of your research team.

I would like to extend my gratitude to Alireza Nemat Saberi, my advisor, for his continuous support, encouragement, guidance and advice throughout the progress of this work. Their extensive expertise in various fields relating to Induction Machine, Condition Monitoring and Machine learning was an inspiration to me.

I also wish to thank Masum Billah, Rajeetharan Sanchayan and my other colleagues at Electromechanics groups as well as my friends Subhadyuti Sahoo and Kirosha Rajasingham for their continuous support and motivation throughout my hard times. Also, I would like to thank all my friends in Finland and Sri Lanka.

Finally, I thank my parents Arumugam Sandirasegaram and Sandirasegaram Krishnadevi and my sisters Anomila Richard and Jenila Elangcheliyan for their support and motivation in completing my thesis. It would not have been possible without them.

Otaniemi, 07.05.2020

Sarvavignoban Sandirasegaram
Contents

Abstract iii
Preface iv
Symbols and Abbreviations vii
List of Figures ix
List of Tables xi

1 Introduction 1
1.1 Background and Motivation ................. 1
1.2 Objective ........................................ 2
1.3 Thesis Structure ................................. 2

2 Faults in Induction Motor and its Condition Monitoring 4
2.1 Types of Faults in Induction Motor .......... 4
2.1.1 Stator Faults ................................. 4
2.1.2 Rotor Faults ................................. 5
2.1.3 Bearing Faults ............................... 6
2.1.4 Eccentricity Faults ......................... 6
2.2 Condition Monitoring Approaches .......... 7
2.2.1 Vibration Monitoring ....................... 7
2.2.2 Motor Current Signature Analysis (MCSA) ... 9
2.3 Chapter Summary .............................. 10

3 Artificial Intelligence 11
3.1 Support Vector Machine (SVM) ............ 11
3.1.1 Hard Margin SVMs ......................... 11
3.1.2 Soft Margin SVMs ......................... 15
3.1.3 Non-Linear SVMs .......................... 16
3.1.4 Design and Tuning ......................... 19
3.1.5 Multiclass SVM ............................ 20
3.2 Artificial Neural Network .................... 20
3.3 Under-fitting, Appropriate-fitting and Over-fitting in ML ......... 21
3.3.1 Under-fitting ............................... 21
3.3.2 Over-fitting ............................... 21
3.4 Cross-validation (CV) .......................... 21
3.5 Dimensionality Reduction .................... 22
3.5.1 Principal Component Analysis (PCA) ....... 22
3.6 Chapter Summary .............................. 24
Symbols and Abbreviations

Symbols

$\alpha$  
Lagarangian multiplier

$\rho$  
Margin

$\xi_i$  
Slack variable

$\mu$  
Dual variable

$a$  
Scaling parameter

$b$  
Bias

$b_1$  
Shifting Parameter

$C$  
Slack Penalty

$d$  
Degree Parameter

$f_1$  
Machine frequency

$f_{bd}$  
Bearing damage frequency

$f_{brb}$  
Broken rotor bar frequency

$f_{ec}$  
Eccentricity frequency

$f_s$  
Supply frequency

$f_{st}$  
Short turn stator winding frequency

$f_v$  
Vibration frequency

$ FD_{1} \rightarrow FD_{12}$  
Frequency Domain Features

$I$  
Current

$K$  
Kernel

$m$  
Number of datapoints

$n$  
Number of samples

$n_d$  
Dynamic integer

$n_{sa}$  
Saturation integer

$n_{ws}$  
Time harmonic integer

$p$  
Pole pairs

$r$  
Distance from the observation $x_i$ to the separator boundary

$r_a$  
Number of rotations

$V$  
Voltage

$w$  
Weight vector

$x_i$  
Input vector

$y_i$  
Output label

Abbreviations

AI  
Artificial Intelligence

BPNN  
Back Propagation Neural Network

CM  
Condition Monitoring

CNN  
Convolutional Neural Network

DAQ  
Data Acquisition

DBN  
Deep Belief Network

DCDP  
Dimensional Characteristic Parameter Distance
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>DE</td>
<td>Dynamic Eccentricity</td>
</tr>
<tr>
<td>DSR</td>
<td>Digital Signal Recorder</td>
</tr>
<tr>
<td>ECN</td>
<td>Edge Computing Node</td>
</tr>
<tr>
<td>EMD</td>
<td>Empirical Mode Decomposition</td>
</tr>
<tr>
<td>F</td>
<td>Full Load</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>FH</td>
<td>Full and Half Load</td>
</tr>
<tr>
<td>FHN</td>
<td>Full, Half and No Load</td>
</tr>
<tr>
<td>FL</td>
<td>Fuzzy Logic</td>
</tr>
<tr>
<td>FS</td>
<td>Feature Selection</td>
</tr>
<tr>
<td>HHT</td>
<td>Hibert-Hung Transform</td>
</tr>
<tr>
<td>IM</td>
<td>Induction Motor</td>
</tr>
<tr>
<td>LR</td>
<td>Logistic Regression</td>
</tr>
<tr>
<td>MCSA</td>
<td>Motor Current Signal Analysis</td>
</tr>
<tr>
<td>ME</td>
<td>Mixed Eccentricity</td>
</tr>
<tr>
<td>MRRR</td>
<td>Minimum Redundancy Maximum Relevance</td>
</tr>
<tr>
<td>OAA</td>
<td>One-Against-All</td>
</tr>
<tr>
<td>OAO</td>
<td>One-Against-One</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Component Analysis</td>
</tr>
<tr>
<td>RPM</td>
<td>Revolutions per minute</td>
</tr>
<tr>
<td>SE</td>
<td>Static Eccentricity</td>
</tr>
<tr>
<td>STFT</td>
<td>Short Time Fourier Transform</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
<tr>
<td>WA</td>
<td>Wavelet Analysis</td>
</tr>
</tbody>
</table>
# List of Figures

1. Types of IM faults and diagnosis techniques ........................................... 4
2. Winding damaged (a) due to overload, (b) due to grounded at edge of slot (c) by voltage surge ................................................................. 5
3. Types of bearing faults (a) Localized defects. (b) Distributed defects .... 6
4. Example of (a) static eccentricity and (b) dynamic eccentricity faults in induction motor .......................................................... 7
5. Dynamic eccentricity rotor vibration signal - Amplitude vs Time ........ 8
6. Dynamic eccentricity rotor Vibration signal - Amplitude vs Frequency . 9
7. SVM linear classification of two classes ............................................... 12
8. $\omega$ and $u$ are perpendicular to $D_1$ .................................................. 13
9. $k$ is a vector of length $d$ perpendicular to $D_2$ .................................. 13
10. Soft Margin SVM classifier .................................................................... 15
11. Non Linear SVM classifier .................................................................... 17
12. The effect of the degree of a polynomial kernel .................................. 18
13. Gaussian Kernel ..................................................................................... 18
14. Sigmoid Kernel ...................................................................................... 19
15. Feed-forward neural network ............................................................... 20
16. Under-fitting Appropriate and Over-fitting in Machine Learning ........ 21
17. Diagram of $k$-fold cross-validation with $k = 10$ ............................... 22
18. PCA workflow ....................................................................................... 22
19. Classification of Data Fusion ................................................................. 25
20. Fault Diagnosis System ......................................................................... 27
21. Schematic representation of the experimental setup ......................... 28
22. Vibration signal of Dynamic eccentricity fault X axis vs Time (Full-Load) 30
23. Sample segmentation signal ................................................................. 30
24. Feature level fusion .............................................................................. 34
25. Feature Rank ....................................................................................... 35
26. (a) PCA classification on operating condition 1, (b) PCA classification on operating condition 2 ......................................................... 37
27. (a) 3-Class full-load training confusion matrix (b) 3 Class full-load testing confusion matrix .......................................................... 38
28. (a) 3 Class full and half-load training confusion matrix (b) 3 Class full and half-load testing confusion matrix .................................. 38
29. (a) 3 Class full-load, half-load and no-load training confusion matrix (b) 3 Class full-load, half-load and no load testing confusion matrix .... 39
30. Faulty classes vs Accuracy ................................................................... 40
31. (a) 8 Class full-load half-load and no-load training confusion matrix (b) 8 Class full-load, half-load and no load testing confusion matrix .... 40
32. (a) 8 Class full-load, half-load and no-load sensor fusion training confusion matrix (b) 8 Class full-load, half and no load sensor fusion testing confusion matrix .......................................................... 41
33. (a) 7 Class-Accuracy versus Number of selected features Accuracy (b) 8 Class - Accuracy versus Number of selected features .......... 42
34  (a) 7 Class- Number of fused sensors  (b) 8 Class - Accuracy versus Number of fused sensors
### List of Tables

<table>
<thead>
<tr>
<th></th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Loading Levels</td>
<td>29</td>
</tr>
<tr>
<td>2</td>
<td>Segmented Data</td>
<td>31</td>
</tr>
<tr>
<td>3</td>
<td>Time Domain Features</td>
<td>32</td>
</tr>
<tr>
<td>4</td>
<td>Frequency Domain Features</td>
<td>33</td>
</tr>
<tr>
<td>5</td>
<td>Fault classes</td>
<td>36</td>
</tr>
<tr>
<td>6</td>
<td>Number of samples</td>
<td>36</td>
</tr>
<tr>
<td>7</td>
<td>Combined samples</td>
<td>37</td>
</tr>
<tr>
<td>8</td>
<td>Operating Conditions</td>
<td>37</td>
</tr>
<tr>
<td>9</td>
<td>Multi class accuracy comparison</td>
<td>39</td>
</tr>
<tr>
<td>10</td>
<td>One sensor versus Multiple sensor fusion</td>
<td>42</td>
</tr>
</tbody>
</table>


1 Introduction

1.1 Background and Motivation

Induction motors (IMs) have gained a prominent place in industrial applications due to their reliability, long-term use, safe operation, low cost and simple design. However, because of their continuous function, faults occur in critical electro-mechanical components. Most of the IM faults appear on the bearing (42%) and stator (28%) parts followed by the rotor (8%) [1]. The faults may trigger shutdowns, device deterioration, noise, overheating, and high maintenance costs [2]. This will cause significant economic and manufacturing losses to industries. Astutely precise machine condition monitoring techniques and proper diagnosis of the machine failure is required to avoid such a problem.

Condition monitoring is an established field playing a significant role in today’s modern industries to protect their investments. The objective of the condition monitoring technique is to identify the unexpected failures or damages in the machine before it causes problems. This approach can provide early warning of potential faults, reduce unintended downtime, cut down maintenance costs and increase stability. There are different types of conventional methods used for fault detection, for example, vibration monitoring [1] - [3], motor current signature analysis (MCSA) [4], magnetic flux, thermal monitoring, noise monitoring and stray flux monitoring. These methods can be used alone or in combination, depending on the parameters needed to identify the machine failures [5].

The role of sensors in conditional monitoring occupies a very prominent place. In the early years, only one kind of sensor was used. Lane Maria et al [6] presented an intelligent mechanical fault diagnostic system using Support Vector Machines (SVM) using only one accelerometer sensor. However, a sensor cannot identify all kinds of faults in a machine; it requires a wide variety of sensors. In this research work, various sensing devices such as vibration and current sensors are mounted at different locations of the machine and provide complementary information on health conditions of the system. Both vibration analysis and MCSA are considered as noninvasive methods in fault diagnosis of rotary machines and many commercialized systems containing vibration and current sensors are employed in industry. Comparison of these two approaches shows that the accuracy of IM fault detection depends on the fault diagnosis method which has been selected. It is reported that MCSA techniques are more effective in detection of electrical faults, while vibration analysis techniques are more precise in dealing with mechanical faults [7]. It is because each type of sensor signal contain specific information depends on the operation condition [8]. However, the obtained data is so large and complex, it requires more trained researchers to analyze the state of the machine and determine its current state.

Artificial Intelligence (AI) technologies were developed to improve this method [9] - [10]. AI approaches have several benefits over traditional methods of fault detection [11]. With continuous research and development over the past years, several AI techniques have been employed to inspect the conditioning monitoring process of machine fault. SVM,
Artificial Neural Networks (ANN) [12], Logistic Regression (LR) [13] and Fuzzy Logic (FL) [14] systems results have been giving precise results. Among different machine learning methods, SVM has been recently considered as one of the most popular ones being used in many research works to train the classifier for fault diagnosis [15]. Soualhi et al [16] used SVM to detect the degradation condition of bearing. Tan et al [17] proposed a support vector machine-based approach for machinery decay status assessment. It has already demonstrated substantial success in medical diagnostics, optical character identification, electrical load estimation and other areas [16].

Sensor fusion technology is capable of handling the confusion and uncertainty of data. Therefore, fusing the current and vibration sensors and monitoring them together can be an efficient method to enhance the performance of fault diagnosis and enable multifault classification simultaneously under complex conditions [18]. This thesis illustrates feature level fusion method for broken rotor bar and eccentricity faults in IM using both current and vibration signals based on SVM method. It also explains the usage of classification strategies for machine learning.

1.2 Objective

The main aim of the thesis is to develop a new, convincing and comprehensive sensor fusion method to solve the established research problem to improve the accuracy of machine fault diagnosis using machine learning techniques.

The following research goals are established

- To extract the inputs required for the machine learning algorithm from the features of the raw data
- To select an effective feature selection method for diagnosing machine fault
- To develop a machine learning algorithm and choose the suitable parameters
- To develop a sensor fusion approach to increase the classification accuracy

1.3 Thesis Structure

Chapter two describes the types of faults that occur in the IM and the condition monitoring approaches.

Chapter three introduces two main AI systems, SVM and ANN. SVM is used as a primary technique in our study, so its types and calculations are described in detail.

Chapter four deals with sensor fusion techniques and their types. This chapter explores examples and advantages of utilizing them in previous researches.

Chapter five explains a process utilizing the support vector machine to classify eight different types of fault in the IM in full-load, half-load, and no-load operating conditions.
Chapter six shows the results obtained from the machine learning algorithm through a combination of full-load, half-load, and no-load operating conditions.

Chapter seven presents the conclusion from the results and suggestions for the future work.
2 Faults in Induction Motor and its Condition Monitoring

Faults in the induction motor have become more frequent in recent times. Section 2.1 describes the main induction motor faults and their causes. Section 2.2 describes the condition monitoring techniques and their methods of selecting parameters.

2.1 Types of Faults in Induction Motor

Faults in an IM can reduce the performance of the machine. The faults occur due to various factors such as low maintenance, production errors and long-term use of the product. Figure 1 indicates different types of faults in an IM and the fault diagnosing techniques. Various techniques are used to detect mechanical and electrical faults. Vibration monitoring and motor current signature analysis techniques are the most widely used. This Figure 1 is referred to as the opinion of the author in [19].

![Diagram of Induction motor fault diagnosis]

---

**Figure 1:** Types of IM faults and diagnosis techniques from [19]

2.1.1 Stator Faults

The stator is one of the significant fault areas in the IM. The stator faults are caused by electrical, mechanical and thermal stresses. They can be categorized as lamination faults and stator winding faults. An IM’s stator has two main parts: 1) the stator core, and 2) the stator windings [20].
Stator Core Faults

The stator core is made of multiple layers of steel laminations. The laminations are placed next to each other and face the same way to make a proper circular layer. An insulation coating is applied to the lamination sheet beforehand to prevent eddy currents from travelling between the sheets. However, sometimes the insulation layer may develop failure which can lead to lamination faults. The lamination faults emerge due to various reasons such as vibration, abrasion, physical damage or faulty insulation coating.

Stator Windings Faults

In an IM, the insulated copper wires on the stator slots are called stator windings. The most common cause of stator winding faults is the failure of the insulation between two adjacent turns on a coil. Stator winding faults are the most dangerous faults and are likely to cause considerable damage to expensive machinery [2]. The most common faults in stator winding are shown in Figure 2.

![Image](image_url)

Figure 2: Winding damaged (a) due to overload, (b) due to grounded at edge of slot (c) by voltage surge

An abrupt temperature change in the IM has a direct impact on the stator windings. Winding insulation can only withstand a specific range of temperatures. Beyond a certain temperature, due to the quality of its material, insulation can start to melt and damage the machine [21].

2.1.2 Rotor Faults

The rotor bar causes mechanical and thermal stress when the rotor is running in low voltage and high temperature. The most frequent rotor faults are broken rotor bars and end-rings, rotor winding short-circuits, and the eccentricity of the rotor bars. The common effects of the rotor broken bar are a reduction in speed and torque, harmonic losses and increase in stator current [22]. Broken rotor bar does not indicate any symptoms at the early stage, but it can do so later because of the propagation of the fault, i.e. more and more broken bars. Therefore, early detection of the rotor faults is highly desirable. Rotor failures are usually 10% of the total IM failures [23].
2.1.3 Bearing Faults

Bearing faults are the industry’s most widespread defects, and the failures are about 40% of the total IMs failures. Bearing faults can arise from various reasons such as incorrect assembly, inappropriate temperature or unstable operation due to oil and whirl. If these failures are not detected at an early stage, then it can lead to higher maintenance costs and wastage of time. The bearing fault of a three-phase squirrel cage IM by using an AI method has been monitored and analyzed in [24]. The findings indicate that measurements taken for a defective motor under load and no-load conditions are radically different from healthy motor reading. Figure 3 shows two types of bearing faults.

![Figure 3: Types of bearing faults (a) Localized defects. (b) Distributed defects from [25].](image)

2.1.4 Eccentricity Faults

An eccentricity fault occur when there is no uniform air gap between the rotor and the stator in the IM [21]. Eccentricity faults are caused by corrosion, mechanical obsolescence and manufacturing error in the bearings [26]. This can create a negative impact on the motor’s output cycle accuracy. Eccentricity faults are classified into three categories: static eccentricity (SE), dynamic eccentricity (DE), and mixed eccentricity (ME).

The static eccentricity is when the center of the geometry and the rotation of the motor are on the same axis, but the center of the stator is different from these two as shown in Figure 4 (a). Static eccentricity causes severe damage to the stator and rotor core windings. The rotor rotation and geometric axes are distinct in dynamic eccentricity shown in Figure 4 (b), but the rotor rotation axis is equal to the stator axis.
2.2 Condition Monitoring Approaches

Modern machinery’s role of condition monitoring and fault diagnosis has become increasingly difficult as machinery becomes more complicated. The following list contains the primary monitoring techniques used in industrial applications.

- Vibration monitoring
- Motor current signature analysis (MCSA)
- Noise monitoring
- Temperature monitoring

The process of the condition monitoring is described hereafter. Initially it collects and stores the specific data from one of the above techniques. The collected data can then be monitored and analyzed when a fault or breakdown might occur. Collecting and monitoring these conditions allow industries to predict when a fault, or even a breakdown, might occur and notify for appropriate maintenance.

2.2.1 Vibration Monitoring

Vibration monitoring technique is most popular than other methods because it is a more efficient technique to obtain information about the internal health conditions of the machine. This monitoring technique involves analyses of vibration signals collected from various sensors are placed at different locations on the machine where the signals are expected to be reliable. Generally, the vibration signals of a machine is low, but in some unforeseen circumstances, a sudden increase in the vibration signals can be observed if something goes wrong in the machine [27]. Numerous signal analysis techniques have been investigated in the past to detect IM faults by using vibration signals such as Fast
Fourier Transform (FFT), Short Time Fourier Transform (STFT), Wavelets analysis (WA), Empirical Mode Decomposition (EMD), Hilbert-Huang Transform (HHT) [28]. However, dealing with these methods requires human intervention, which is a major challenge. Thus, in recent times vibration levels are monitored using AI techniques.

Vibration signal analysis can be classified into three parts (a) time domain, (b) frequency domain and (c) time-frequency domain. Through the time domain analysis waveform, orbits and statistical parameters can be detected [3].

The time domain vibration signals of dynamic eccentricity rotor measured in section 5.2 is shown in Figure 5.

![Amplitude vs Time](image_url)

Figure 5: Dynamic eccentricity rotor vibration signal - Amplitude vs Time

**Spectrum Analysis**

The FFT has obtained the spectrum of the vibration signal. The spectrum is generated by the time wave-forms from the machine. Each component in the machine generates a single frequency, which can be easily identified from the spectrum.

The frequency spectrum of Figure 5 is shown in Figure 6. Amplitude and frequency are the two main components used to analyze the vibration signal. Vibration amplitudes show the seriousness of the fault and the frequencies of vibration display the origin of the fault. The harmonics and side-band in the spectrum show the motion and related forces within the machine. The peaks appear to indicate the sources for the vibration of all the specified frequencies.
2.2.2 Motor Current Signature Analysis (MCSA)

MCSA is a method of detecting faults in the IM by analyzing the stator current in the motor. In the event of any faults inside or outside of the motor, its impact is exposed to the stator current. This method is particularly useful in the industries where the motor does not have direct access. Based on previous researches [29, 30], the speed, torque, flux, and rotor current can be calculated using the MCSA to monitor the motor’s health condition [31]. Typically, the frequency spectrum can detect distinct patterns characteristic of different faults in the motor.

MCSA can detect the following electrical machine faults.

1. Broken Rotor Bars

The frequencies of the motor current in the broken rotor bar are given by equation (1).

\[ f_{brb} = f_s \left[ k \left( \frac{1 - s}{p} \right) \pm s \right] \]  

\( f_s \) = supply frequency, \( p \) = pole pairs, \( s \) = slip, \( k = 1, 3, 5 \ldots \)
2. Air Gap Eccentricity

The following equation (2) shows the frequency components of the high current generated due to an abnormal air gap between the rotor and the stator

\[ f_{ec} = f_s \left\{ (r_a \pm n_d) \left( \frac{1 - s}{p} \right) \pm n_{ws} \pm 2n_{sa} \right\} \quad (2) \]

\( n_d = \) dynamic integer ±1 , \( n_{ws} = \) time harmonic integer 1,3,5,7,... , \( n_{sa} = \) saturation integer, \( r_a = \) number of rotor bars

3. Bearing Damages

The radial displacement between the stator and the rotor is formed by mechanical vibration, which can lead to bearing damages. As a result, the following equation (3) shows the frequencies in line current.

\[ f_{bd} = (f_s \pm m \cdot f_v) \quad (3) \]

\( f_v = \) bearing fault characteristic frequencies , \( m = 1,2,3,\ldots \)

4. Shorted Turns in Stator Windings

Frequencies induced in the rotor current due to inter-turn fault are shown in equation (4). Short turns cause excessive heat and current imbalance in the stator windings.

\[ f_{st} = f_s \left\{ \frac{n}{p} (1 - s) \pm k \right\} \quad (4) \]

\( k = 1,2,3,\ldots \)

2.3 Chapter Summary

Induction motor faults cannot be avoided. With this notation, major faults are in the electrical and mechanical components of the induction motor were discussed in Section 2.1. In Section 2.2, the condition monitoring procedure of induction motor using current and vibration signals were analyzed. The main artificial intelligence techniques for machine condition monitoring are discusses in the next chapter.
3 Artificial Intelligence

AI is the broader concept of science, imitating human abilities. Machine learning (ML) is a subset AI, which provides systems with the ability to learn and develop automatically from experience without being expressly programmed [32]. There are different algorithms (e.g. support vector machine, neural network) in ML that help to solve problems. ML is an effective approach of machine condition monitoring and fault diagnosis. In this chapter, the theoretical foundation of ML techniques such as SVM and ANN are briefly presented. Section 3.1 discusses theory, structures and tuning parameters of SVM. The working principles of ANN are presented in detail in Section 3.2. In ML, the cause of a model’s poor performance is either under-fitting or over-fitting briefly discusses in Section 16. Cross-validation (CV) is commonly used to determine the effectiveness of the ML model. Section 3.4 describes the CV procedure. Section 3.5 illustrates the process of dimensionality reduction.

3.1 Support Vector Machine (SVM)

SVM is a supervised machine learning technique introduced by Vapnik and his co-workers in the mid of 1990s [33]. The main idea behind SVM is to draw a hyperplane in n-dimensional space such that it maximizes the margin between classification groups. The SVM is generally designed for 2-class problems. This is binary classification, where one class is the positive class and the other one is the negative. This method separates two different classes of data by choosing suitable boundary line or hyperplane.

3.1.1 Hard Margin SVMs

For instance, let us assume we have a training dataset for binary classification problem: \((x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)\) with \(x_i \in \mathbb{R}^n\) and \(k\) being the total number of training samples. The output labels consist of positive \((y_i = +1)\) and negative \((y_i = -1)\) labels. As shown in Figure 7a, there are many possible ways to separate the two classes. However, to select the optimal one, the line must have the maximal distance (margin) between the two classes’ adjacent data points. These points are called support vectors.

The following equation (5) expresses the boundary line \(D_0\), which attempts to classify the two classes.

\[
f(x_i) = (w^T x_i + b),
\]  

where \(x_i\) represents the input vector, \(w \in \mathbb{R}^n\) is a weight vector , and \(b\) is a scalar threshold. When all of the given datasets implement their constraints

\[
\begin{align*}
    w^T x_i + b & \geq +1 \quad y_i = +1 \\
    w^T x_i + b & \leq -1 \quad y_i = -1
\end{align*}
\]  

(6)

To satisfy all the constraint, the above equation (6) can be combined into

\[
y_i(w^T x_i + b) \geq 1 \quad \forall i
\]  

(7)
As $x_0$ is in $D_2$, the margin ($d$) is the distance between two hyperplane $D_1$ and $D_2$. The primary target of the Hard Margin SVM is to maximize the margin and place the boundary line between two classes.

The distance from the observation $x_i$ to the separator boundary can be defined as

$$r = \frac{|w^T x_i + b|}{\|w\|} \quad (8)$$

In case if $d$ is a scalar, adding $d$ to $x_0$ won’t create a new hyperplane. Because $x_0$ is a vector and $d$ is a scalar, adding a scalar with a vector is impossible. However, adding two vectors is possible, so the best way is to transform $d$ into a vector so we can able to do an addition. According to equation 6 we know vector $w$ is perpendicular to $D_1$.

First, multiply $u$ by $d$ to get a new vector $k$

$$k = du \quad (9)$$

Where $u = \frac{w}{\|w\|}$, the unit vector of $w$ As it is a unit vector $\|u\| = 1$, and it has the same direction as $w$ as shown in Figure 8, so it is also perpendicular to the hyperplane.
From the equation 9 the following properties can be obtained

1. $||k|| = d$

2. $k$ is perpendicular to $D_2$ (because it has the same direction as $u$)

from these above properties we can conclude that $k$ is a vector as shown in Figure 9

Equation 10 transform scalar $d$ into a vector $k$ which can use to perform and addition with the vector $x_0$.

$$k = d\mathbf{u} = d\frac{\mathbf{w}}{||\mathbf{w}||}$$  \hspace{1cm} (10)

When $y_i \in \{-1, 1\}$

$$r = \frac{y_i \mathbf{w}^T \mathbf{x}_i + b}{||\mathbf{w}||}$$  \hspace{1cm} (11)

Now, by considering the constraint $y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1$ for each observation $i$, we can maximize the distance (margin) by minimizing $||\mathbf{w}||$. 

Figure 8: $\mathbf{w}$ and $\mathbf{u}$ are perpendicular to $D_1$

Figure 9: $k$ is a vector of length $d$ perpendicular to $D_2$
\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|w\|^2 \\
\text{subject to} & \quad y_i((w^T x_i) + b) \geq 1
\end{align*}
\]

(12)

Now this is a constraint optimization problem. Therefore, Lagrangian multiplier \(\alpha_i\) are introduced as follows:

\[
L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^{N} \alpha_i (y_i((w^T x_i) + b) - 1)
\]

(13)

The SVM convex optimization problem is defined as:

\[
\begin{align*}
\text{minimize} & \quad L(w, b, \alpha) \\
\text{subject to} & \quad \alpha_i \geq 0, \ i = 1, \ldots, N
\end{align*}
\]

(14)

The goal is to reduce the equation (13) with respect to \(w\) and \(b\), thus allowing the derivatives of \(L\) and \(\alpha\) to disappear.

\[
\begin{align*}
\frac{\partial L}{\partial w} &= w - \sum_{i=1}^{N} \alpha_i y_i x_i = 0 \\
\frac{\partial L}{\partial b} &= \sum_{i=1}^{N} \alpha_i y_i = 0
\end{align*}
\]

(15)

Since this is a primal optimization problem, SVM solves this by a dual optimization method followed as:

\[
\begin{align*}
\text{maximize} & \quad (L(\alpha_i) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=0}^{N} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \\
\text{subject to} & \quad \sum_{i=1}^{N} \alpha_i y_i = 0 \\
& \quad \alpha_i \geq 0, \ i = 1, \ldots, N
\end{align*}
\]

(16)

\(\alpha_i\) has been obtained from equation (16). From that maximum margin \(w\) has been calculated as follows:

\[
w = \sum_{i=1}^{N} \alpha_i y_i x_i
\]

(17)

After solving the dual problem the decision boundary is defined as:

\[
f(x_i) = \sum_{i=1}^{N} \alpha_i y_i (x_i \cdot x_j) + b
\]

(18)
3.1.2 Soft Margin SVMs

So far we implicitly assumed our datasets is linearly separable. However, most of the real datasets are noisy and cannot be separated without error. To handle this case, soft margin SVM is developed with relaxed constraints in which certain number of data points are allowed to lie beyond the boundary [33]. It deals with two primary procedures. It allows a certain number of data points to lie beyond the boundary and keep the margin as wide as possible.

\[ y_i (w^T x_i + b) \geq 1 - \xi_i \quad \xi_i \geq 0 \quad \forall i \]  

(19)

The value of the slack variable (\( \xi_i \)) is the distance between the margin and the other side of the misclassified data-point.

![Diagram of Soft Margin SVM classifier](image)

Figure 10: Soft Margin SVM classifier

It can be observed from the Figure 10 that when \( \xi_i = 0 \), the observation \( i \) is classified correctly and it is located on the right side of the hyperplane and outside of the margin. When \( 0 < \xi_i < 1 \), the observation \( i \) is classified correctly and is within the margin and when \( \xi_i > 1 \), the observation \( i \) is misclassified and it is on the wrong side of the hyperplane.

The soft margin SVM optimization problem can be formulated as follows:

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{N} \xi_i \\
\text{subject to} & \quad y_i ((w^T x_i) + b) \geq 1 - \xi_i \quad \xi_i \geq 0
\end{align*}
\]

(20)

In the above equation (20) multiply the sum of slack variables by the parameter \( C \), which is called a slack penalty. It controls the maximizing of the margin and minimizing of the training error. In this section there is a new constraint \( \xi_i \geq 0 \), therefore another new variable \( \mu_i \) has been added.
Lagrangian function

\[ L(w, b, \alpha, \xi, \mu) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \alpha_i (y_i((w^T x_i) + b) - 1) - \sum_{i=1}^{N} \xi_i \mu_i \]  

(21)

the derivatives are taken as:

\[
\begin{align*}
\frac{\partial L}{\partial w} &= w - \sum_{i=1}^{N} \alpha_i y_i x_i = 0 \\
\frac{\partial L}{\partial b} &= \sum_{i=1}^{N} \alpha_i y_i = 0 \\
\frac{\partial L}{\partial \xi_i} &= C - \alpha_i - \mu_i = 0
\end{align*}
\]

(22)

To find the dual problem

\[ \text{minimize} \quad L(w, b, \xi, \alpha) \quad \xi_i \geq 0 \]  

(23)

\[ \text{maximize} \quad L(\alpha_i) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=0}^{N} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \]  

subject to

\[ \sum_{i=1}^{N} \alpha_i y_i = 0 \]

\[ C \geq \alpha_i \geq 0, \quad i = 1, \ldots, N \]  

(24)

Overall, the only difference is that C has an additional upper bound on the soft edge compared with hard-margin SVM.

3.1.3 Non-Linear SVMs

So far the linear SVM model is introduced. In many cases including the application of fault diagnosis of rotary machines, even if the hyperplane is determined to be optimal, the training data cannot be appropriately classified by linear SVM due to its high nonlinearity. To deal with this type of problems, as shown in Figure 11 the original space can be transferred to a high dimensional space \((x \rightarrow \phi(x))\), where it can be classified by linear classification method [34].
However, computing $\phi(x_i, x_j)$ is very complicated and time-consuming. To overcome this, the kernel function is introduced as $K(x_i, x_j) = \phi(x_i) \phi(x_j)$. A kernel is a similarity function; it takes two inputs and outputs their similarity. The linear decision boundary in high dimensional space is given by equation 25, where the input vector $x_i$ is transferred to high dimensional space $\phi(x_i, x_j)$.

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i K(x_i, x_j) + b$$

(25)

$x_i$ is the $i^{th}$ training input and $x_j$ is the unlabeled input. There are a number of common kernel choices; given below are few examples of the kernel function value for a single scalar input feature $x$ to show how the kernel measures the similarity in the original space.

### Polynomial Kernel

The polynomial kernel equation is shown in equation 26, where "d" is degree of the kernel, and "b" is a constant that allows trading the influence of lower-order and higher-order terms.

$$K(x_i, x_j) = (x_i^T x_j + b)^d \text{ for any } d > 0$$

(26)

As shown in Figure 12, the polynomial kernel of degree 1 leads to a linear separation (A). Higher-degree polynomial kernels allow a more flexible decision boundary (B,C) [35].
Gaussian kernel (RBF)

The most common kernel function is the Radial Basics Function (RBF) or Gaussian similarity kernel. The effect of this kernel is high values close to \( x=0 \), dropping off as a Gaussian with some spread sigma when going away from that point. The parameter sigma can be used to control over- and under-fitting. As the sigma changes, its distribution varies shown in Figure 13.

\[
K(x_i, x_j) = \exp\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right) \quad (27)
\]

Figure 13: Gaussian Kernel
Sigmoid Kernel

According to the sigmoid function equation 28, "a" is a scaling parameter, and "b" is a shifting parameter. As shown in Figure 14 when "a" decreases the slope of the graph also decreases, similarly when "b" decreases, the graph moves to the x-direction.

Kernels provide significant strength to support vector machines and the systems with small to moderate numbers of data. At the same time, RBF's are probably the most common choice- a dominant feature of support vector machines that can decide similarity functions based on the task.

\[ K(x_i, x_j) = \tanh(ax_i, x_j + b) \]  

(28)

![Figure 14: Sigmoid Kernel](image)

3.1.4 Design and Tuning

Two essential aspects need to be considered in the SVM design process.

1. Setting error penalty (C)

2. Choosing Kernel function (K)

The size of the margin varies when the value of the error penalty changes. When \( C = \infty \), the margin gets small, and most of the slack variables become zero. It triggers the SVM during the training process, minimizing the classification error and classifying the training points with the best boundary line. However, when using a test set, it does not provide a valid boundary line. The opposite of that (if we change the \( C = 0 \)) the margin gets big, so we ignore the misclassification. Similarly, Kernel function also dominates the choice of the borderline. Depending on the type of kernel used, the boundary line results can vary. When selecting a kernel, we need to consider the tuning parameters of each kernel. Generally, the RBF kernel is often used to choose the most suitable boundary line. In the RBF kernel, \( \sigma \) defines how far the influence of a single data point can reach.
These two parameters $C$ and $\sigma$ are called hyperparameters. Finding optimal hyperparam-eter is a challenging task. To simplify this process, the 10-fold cross-validation (CV) and grid search methods are used. Different pairs of $(C, \sigma)$ values are tried and tested for choosing the best setting having the highest CV accuracy.

### 3.1.5 Multiclass SVM

SVM is generally a two-class classifier, but in real-life problems, there are more than two classes; for that Multiclass SVM is introduced. SVM multiclass is classified into two types One-Against-All (OAA) and One-Against-One (OAO). In the OAA method, one class trains with other classes. For example, if there are four types: Class (A, B, C, D). Class A will be trained with other classes in which class A will be labelled as positive, and all other classes will be labelled negative. In OAO mode, one class will train with the other class. For example, Class (A & B), Class (A & C), Class (B & C) likewise.

### 3.2 Artificial Neural Network

Artificial Neural Network (ANN) is a computational model based on the composition and operation of a biological neural network. Figure 15 shows the structure of a feed-forward neural network. The neural network contains three layers: an input layer, output layer and hidden layer. The information is delivered to the system through the input layer. The forecast results are delivered in the output layer. The layer between input and output is called a hidden layer. Each layer is made up with several smaller units called neurons and each neuron is connected to another neuron by weighted connections. The ANN-based monitoring system can be divided into three parts: training, verification and monitoring. The training of the ANN involves solving the models allowed by several related methods. Verification refers to checking the performance of the model during the training process. An ANN can be configured for recognizing patterns through a learning process, even though the data contained in those patterns are noisy, sparse or incomplete. The advantage of ANN is that after the learning phase, it could respond in a desirable way to an input pattern. According to past researches, it has been shown that ANN is given more importance in detecting faults of IMs because it has a better mathematical model for solving nonlinear problems [36, 37].

![Figure 15: Feed-forward neural network](image-url)
3.3 Under-fitting, Appropriate-fitting and Over-fitting in ML

When designing a machine-learning model, it is considered to be more efficient if it correctly classifies new information. However, the built model can exhibit poor performance in some cases. Accordingly, the model is built to be either too simple or too complicated, so it is difficult to predict the correct output. Figure 16 shows three major factors.

![Diagram of Under-fitting, Appropriate-fitting, and Over-fitting in Machine Learning](image)

Figure 16: Under-fitting Appropriate and Over-fitting in Machine Learning

3.3.1 Under-fitting

As shown on the left side of the Figure 16, the line does not cover all relevant data points. As a result, the model was unable to learn enough patterns from the training data, resulting in low-performance.

3.3.2 Over-fitting

Over-fitting refers; the line covers all the possible points as shown at the right side of the Figure 16. It makes it more effective in the training set but less accuracy with the testing. An important preventive mechanism against over-fitting is cross-validation.

3.4 Cross-validation (CV)

CV is a valuable technique used on training dataset to validate the machine learning models. Through the utilization of CV, the over-fitting of the machine learning model is checked during the training process. In k-fold CV, k separate learning experiments are conducted. One of these k subsets is chosen as a test set, and the model is trained using the remaining k-1 sets as training data. The average error over all k-folds is then determined.

The value of k should be chosen carefully according to the dataset. If k is chosen incorrectly, it can lead to a misconception about the capabilities of the model. When the k value is high, only a small number of sample combinations are possible. It leads to low bias, high variance and high computational time for the model. Similarly, when the value for k is low, the model becomes low variance and high bias. In this research, a 10-fold CV
is selected, as shown in Figure 17. It refers to ten times iterated, and its average value was obtained.

Figure 17: Diagram of k-fold cross-validation with k = 10 from [38]

3.5 Dimensionality Reduction

3.5.1 Principal Component Analysis (PCA)

PCA is a dimensionality reduction or data compression technique that identifies the correlations between patterns among different features in the dataset. The main objective of the PCA is to reduce high dimension data to low dimension data while maintaining the maximum variation of the original dataset. This approach can also be used for feature selection (FS). PCA workflow is shown in Figure 18.

Figure 18: PCA workflow
Consider a two-dimension ($m \times n$) data matrix, $X$, where $m$ is the number of samples, and $n$ is the number of features. PCA transforms the matrix $X$ into new coordinate's $E$ using transformation matrix.

$$E = TX$$  \hspace{1cm} (29)

In the above equation (29), $T$ is the transformation matrix, and $E$ is the matrix of principal component. As shown in the Figure 18, the process of PCA is explained in detail in the next paragraph.

**Standardization of the data**

In the data analysis, the initial variables are usually found in different ranges. The variables with larger range have an almost obvious impact on the output. To overcome this scale-dependency, standardization of the data scales all the variables and their values and helps to lie them within a similar range.

$$Z = \frac{\text{Variable value} - \text{mean}}{\text{Standard deviation}}$$  \hspace{1cm} (30)

After performing the standardization, all the variables are scaled across a standard and a comparable scale. Transformation matrix $T$ is essentially designed to maximize the variance and minimize the covariance.

**Computing the covariance matrix**

Covariance matrix helps to identify the correlations between the different variables in a dataset. Covariance between two variables is an indication of whether it changed together or in the opposite direction. The covariance matrix of $Z$ is shown in equation 31.

$$C = Z^T Z = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \bar{X})(X_i - \bar{X})^T$$  \hspace{1cm} (31)

Where $n$ is number of samples, $X_i$ is the $i^{th}$ raw score in the first set of samples, $\bar{X}$ denotes the means of $X$.

**Calculate the eigenvectors and eigenvalues**

$$Z^T Z = PD P^{-1}$$  \hspace{1cm} (32)

In the above equation 32, $P$ represents a matrix of eigenvectors, and $D$ is a diagonal matrix. The eigenvalues in the diagonal of $D$ are similar to the columns in $P$. The first element of the eigenvalue in the diagonal of $D$ is $\lambda_1$, and the corresponding eigenvector is first column of $P$. The next step is to sort the eigenvalues $\lambda_1$ to $\lambda_p$ from largest to smallest. Now the sorted matrix is represented by $P^*$.

Calculate $$Z^* = ZP^*$$  \hspace{1cm} (33)

From that, the first few elements containing more information should be select, and the lesser-known elements are ignored. Through the transformation matrix, the number of dimensions is reduced.
3.6 Chapter Summary

Section 3.1 briefly introduced the basic principles of SVM involved in this research, i.e. Hard Margin SVMs, Soft Margin SVMs and Non-Linear SVMs. Section 3.2 provided an overview of the working principles of ANN. Section 3.3 discussed in detail the over-fitting and under-fitting, which contributes to poor machine learning efficiency. Section 3.4 described the CV procedure. The process of dimensionality reduction was illustrated in Section 3.5. The next chapter discusses sensor fusion techniques.
4 Sensor Fusion

Sensor fusion technique is the process of combining data from multiple sensor sources to build more sophisticated decision-making system. This technique can be carried out at three levels: data level, feature level and decision level are described in Section 4.1, Section 4.2 and Section 4.3 respectively. Classification of data fusion is shown in Figure 19.

![Classification of Data Fusion](image)

Figure 19: Classification of Data Fusion from [39]

4.1 Data Level Fusion

Data level fusion method takes place at the data level, where the combination of raw data from multiple sensors is directly sent to the fusion center. The new dataset contains more reliability, precision and valuable information than the raw inputs. This method has two steps, data acquisition and data preprocessing.

In literature [40], an improved data level degradation-based prognostic model has been developed and analyzed with the systematic approach of data selection, data processing and data fusion. A Convolutional Neural Network (CNN) based multi-sensor data-level fusion technique has been used in [41], to diagnosis the fault of rotating machinery. The results show that a better and more accurate fault diagnosis could be obtained from this method. Yan[42] proposed an n-dimensional characteristic parameters distance (n-DCPD) for rolling bearing fault diagnosis. In this method vibration and acoustic emission signals are used for the data level fusion and extracted single feature fusion for the fault diagnosis. This method promises to identify unknown errors in a simple, intuitive and
precise manner. Compared to other techniques, this method could achieve maximum performance due to the loss of less data relative to the other two groups [42].

4.2 Feature Level Fusion

Feature fusion method helps to reduce extensive feature data to small and make it easy to disclose the unseen fault. Gang et al [43] designed an Edge Computing Node (ECN) for real-time fault diagnosis and dynamic control of the rotating machine. Forty features were extracted and fused by the classical Back-Propagation Neural Network (BPNN) method. The results show within a limited period; this method can detect the entire fault with 100% classification accuracy. Zhuyun et al [44] proposed a new approach to multisensor feature fusion for bearing fault diagnosis using Sparse Autoencoder (SAE) and Deep Belief Network (DBN). Features obtained from multiple sensor signals were used as the input of several two-layer SAEs for feature fusion. The experimental results indicate that the proposed method has a higher identification level than other approaches and is less responsive to training samples. Jiejunyi et al [45] proposed a new multi-segment classification approach for rotating machinery based on feature fusion. A Deep Belief Network (DBN) has been developed to minimize the higher dimensions of the extracted features and enhance computational efficiency and accuracy.

4.3 Decision Level Fusion

Decision level fusion determines fault pattern according to the results from multiple classifiers. This is one of the dynamic methodologies of fusion since the fusion takes place in the final stage after the information is fully processed in the initial stages. This proposed method could correctly distinguish between different types of faults. The input of decision-level information fusion is transformed into a simple task of pattern recognition that common supervised learning algorithms can effectively solve.

Duy et al [46] proposed a signal-based motor current fault diagnostic method that could be applied to external bearings in rotary machine systems using deep learning and information fusion. A new decision-level information fusion strategy is used to merge data from all of the convolutional neural networks used to improve classification accuracy. The experimental results show this proposed method has much better performance and cost-effectiveness compared to other existing works. For the diagnosis of electrical and mechanical faults in IMs, a two-stage Bayesian process and Principal Component Analysis (PCA) sensor fusion technique has been proposed in [47]. It has been shown that the algorithm can detect stator faults, broken rotor bar faults, and IM bearing faults. The common challenge facing the development of decision-level fusion algorithm is that there are often many features relative to the number of observations.

4.4 Chapter Summary

In this chapter, a comparative analysis of different types of sensor fusion techniques was discussed. The next chapter describes in detail the methods and techniques used in this research work.
5 Research Methodology

The procedure of the developed fault detection and classification scheme of this thesis is illustrated in Figure 20. Each of the parts is explained in detail from Section 5.1 to Section 5.7.

Figure 20: Fault Diagnosis System
5.1 Experimental Setup

Figure 21 shows the schematic diagram for the measurement setup, which was designed for IM fault detection [48]. The loading machine and testing machines were connected with a back-to-back coupling through their shafts. The loading machine was a three-phase, four-pole squirrel cage IM. This system was fixed to standard mechanical support to help prevent the undesirable effects of vibration on the testing machine. An auto-transformer supplied the input signals through a frequency converter to the loading machine. The output of the frequency converter changed the rotational speed of the loading machine to provide different loading levels.

![Schematic representation of the experimental setup from [48]](image)

5.2 Data Acquisition

Data acquisition (DAQ) is the method of measuring electrical or physical quantities such as voltage, current, temperature and noise with a Digital Signal Recorder (DSR). Preprocessing is the next step where the signals from the sensors are compressed and the noise and other disturbances are eliminated. DSR was used to record and store all the measured signals in the computer. Five tri-axial accelerometers measured the acceleration of the testing machine in three perpendicular axes (x, y, and z). Voltage (in three phases), current and rotational speed were also measured during the experiment. The measured conditions of the machine are as follows:

1. Dynamic eccentricity rotor
2. Dynamic eccentricity two non-consecutive broken bars loose bearing rotor
3. Healthy rotor
4. One broken bar rotor
5. Three broken bar rotor
6. Two broken bar rotor
7. Two non-consecutive broken bars dynamic eccentricity rotor
8. Two non-consecutive broken bars no eccentricity rotor

5.3 Data Preprocessing

As most sensors are located in various sections of the motor, certain external factors, noise and instrument failures can affect sensitive information or include redundant information in the dataset. Preprocessing technique helped to eliminate redundant information, and improve the algorithm. So, it can be able to produce better results with high accuracy. Also, the measured data by the data acquisitions was vast and complex; various issues can be encountered when processing it directly. Therefore, the signal segmentation method was proposed to reduce complexity and the arithmetical errors. The data obtained has been modified from 1-D matrix to 2-D matrix to conduct the segmentation. The size of the 2-D matrix was organized by m x n matrix, where m is the number of data points and n is the number of samples.

Table 1: Loading Levels

<table>
<thead>
<tr>
<th></th>
<th>No-Load Steady State</th>
<th>Half-Load Steady State</th>
<th>Full-Load Steady State</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Phase current (A)</strong></td>
<td>18</td>
<td>30</td>
<td>40</td>
</tr>
<tr>
<td><strong>Phase voltage (V)</strong></td>
<td>190</td>
<td>190</td>
<td>190</td>
</tr>
<tr>
<td><strong>Line to Line voltage</strong></td>
<td>325</td>
<td>325</td>
<td>325</td>
</tr>
<tr>
<td><strong>Measured machine RPM</strong></td>
<td>1497</td>
<td>1447</td>
<td>1421</td>
</tr>
<tr>
<td><strong>Drive RPM</strong></td>
<td>-</td>
<td>1435</td>
<td>1405</td>
</tr>
<tr>
<td><strong>Frequency (Hz)</strong></td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
</tbody>
</table>

Measured machine RPM ::- Real RPM measured by rotational speed sensor.
Drive RPM ::- Value set to the drive-loading machine fed by frequency converter control software.

As shown in Table 1, there were three operational conditions used in this experiment. For each condition of the machine, the RPM barely change. Thus, a common RPM value of 1497 rpm from no-load condition was chosen.

The number of data points of the segment was calculated as below:

\[
\text{Sampling Frequency} \quad f_s = 10,000 \text{Hz} \quad (34)
\]

\[
\text{Machine Frequency} \quad f_1 = 50 \text{Hz} \quad (35)
\]

\[
\text{One electrical period} \quad \left(\frac{f_s}{f_1}\right) = 200 \text{ data points} \quad (36)
\]

\[
\text{One segment(4 electrical period)} = 4 \times 200 \quad \text{data points} = 800 \quad \text{data points} \quad (37)
\]
The number of data points at each sample can be a hyperparameter, and we can study its effect on the final result (accuracy). Choosing its value for our initial guess is arbitrary. We selected 800 data points since it is equal to 4 complete electrical periods, and we can extract the time-domain and frequency-domain features by FFT easily. After doing that, the accuracy is so high, so it seems that we don’t need to select other numbers for it.

It is also good to check the effect of changing this number on the results in future research. Currently, we have 290*800 data points. It can also be 145*1600 data points or 116*2000 data points or anything else. We can do it for different values and draw a plot showing the variations of total accuracy with respect to the number of data points at each sample.

![Figure 22: Vibration signal of Dynamic eccentricity fault X axis vs Time (Full-Load)](image)

![Figure 23: Sample segmentation signal](image)

Table 2 shows the total number of data for each fault type obtained by the experimental setup which was divided into 800 data points. The samples were calculated and the total length (l) of each fault data was adjusted and converted to (l x m) 2-D matrices. A total of 35 features were extracted from the time and frequency domain from each sample. The process of feature extraction and the feature selection are described in the next section.
Table 2: Segmented Data

<table>
<thead>
<tr>
<th>Fault type</th>
<th>OC</th>
<th>Length of Data</th>
<th>l</th>
<th>m</th>
<th>Adjusted length of data</th>
<th>(l x m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic eccentricity</td>
<td>N</td>
<td>131948</td>
<td>800</td>
<td>164</td>
<td>131200</td>
<td>800x164</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>242756</td>
<td>800</td>
<td>303</td>
<td>242756</td>
<td>800x303</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>232453</td>
<td>800</td>
<td>290</td>
<td>232000</td>
<td>800x290</td>
</tr>
<tr>
<td>Dynamic eccentricity two non-consecutive broken bars loose bearing</td>
<td>N</td>
<td>184148</td>
<td>800</td>
<td>230</td>
<td>184000</td>
<td>800x230</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>264734</td>
<td>800</td>
<td>330</td>
<td>264000</td>
<td>800x330</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>410921</td>
<td>800</td>
<td>513</td>
<td>410400</td>
<td>800x513</td>
</tr>
<tr>
<td>Healthy rotor</td>
<td>N</td>
<td>61271</td>
<td>800</td>
<td>76</td>
<td>60800</td>
<td>800x76</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>100898</td>
<td>800</td>
<td>126</td>
<td>100800</td>
<td>800x126</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>117215</td>
<td>800</td>
<td>146</td>
<td>116800</td>
<td>800x146</td>
</tr>
<tr>
<td>One broken bar rotor</td>
<td>N</td>
<td>125873</td>
<td>800</td>
<td>157</td>
<td>125600</td>
<td>800x157</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>150182</td>
<td>800</td>
<td>187</td>
<td>149600</td>
<td>800x187</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>126872</td>
<td>800</td>
<td>158</td>
<td>126400</td>
<td>800x158</td>
</tr>
<tr>
<td>Three broken bars rotor</td>
<td>N</td>
<td>60272</td>
<td>800</td>
<td>75</td>
<td>60000</td>
<td>800x75</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>50282</td>
<td>800</td>
<td>62</td>
<td>49600</td>
<td>800x62</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>66266</td>
<td>800</td>
<td>82</td>
<td>65600</td>
<td>800x82</td>
</tr>
<tr>
<td>Two broken bars rotor</td>
<td>N</td>
<td>58940</td>
<td>800</td>
<td>73</td>
<td>58400</td>
<td>800x73</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>93905</td>
<td>800</td>
<td>117</td>
<td>93600</td>
<td>800x117</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>161504</td>
<td>800</td>
<td>201</td>
<td>160800</td>
<td>800x201</td>
</tr>
<tr>
<td>Two non-consecutive broken bars dynamic eccentricity rotor</td>
<td>N</td>
<td>280718</td>
<td>800</td>
<td>350</td>
<td>280000</td>
<td>800x350</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>238094</td>
<td>800</td>
<td>297</td>
<td>237600</td>
<td>800x297</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>233099</td>
<td>800</td>
<td>291</td>
<td>232800</td>
<td>800x291</td>
</tr>
<tr>
<td>Two non-consecutive broken bars no eccentricity rotor</td>
<td>N</td>
<td>208790</td>
<td>800</td>
<td>260</td>
<td>208000</td>
<td>800x260</td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>230768</td>
<td>800</td>
<td>288</td>
<td>230400</td>
<td>800x288</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>266752</td>
<td>800</td>
<td>333</td>
<td>266400</td>
<td>800x333</td>
</tr>
</tbody>
</table>

5.4 Feature Extraction

The feature extraction is the process of extracting the essential aspects of measured data without distorting it. This method is the most crucial part of the classification process. It helps to produce the relevant features and makes the classification process faster and smoother. The amount of features determines the accuracy of classification and they should be carefully measured. In this research work, 23 ordinary time-domain features and 12 frequency domain features were selected from each of the vibration/current signals. These features are listed in Table 3 for the time domain and Table 4 for the frequency domain. The main purpose of this study is to increase the diagnostic information of the features in spite of choosing simple and straightforward features. Therefore, all the features obtained from different sensors were concatenated leading to a combined list of 630 features for each sample of our datasets.
### Table 3: Time Domain Features

<table>
<thead>
<tr>
<th>Features</th>
<th>Expressions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>$T_m = \frac{1}{N} \sum_{i=1}^{N} x_i$</td>
</tr>
<tr>
<td>Median</td>
<td>$T_{med} = med(x_i)$</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>$T_{kurt} = \frac{1}{N} \sum_{i=1}^{N} \frac{(x_i - \mu)^4}{\sigma^4}$</td>
</tr>
<tr>
<td>RMS</td>
<td>$T_{rms} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_i)^2}$</td>
</tr>
<tr>
<td>Maximum</td>
<td>$T_{max} = max(x_i)$</td>
</tr>
<tr>
<td>Skewness</td>
<td>$T_{Skew} = \frac{1}{N} \sum_{i=1}^{N} \frac{(x_i - \mu)^3}{\sigma^3}$</td>
</tr>
<tr>
<td>Minimum</td>
<td>$T_{min} = min(x_i)$</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>$T_{Std} = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} (x_i - T_m)^2}$</td>
</tr>
<tr>
<td>Variance</td>
<td>$T_{var} = \frac{1}{N-1} \sqrt{\sum_{i=1}^{N} (x_i - T_m)^2}$</td>
</tr>
<tr>
<td>Square root amplitude</td>
<td>$T_{sra} = \left(\frac{1}{N} \sum_{i=1}^{N}</td>
</tr>
<tr>
<td>Peak to Rms ratio</td>
<td>$T_{peak2rms} = \frac{</td>
</tr>
<tr>
<td>Amplitude Square</td>
<td>$T_{as} = \frac{1}{N} \sum_{i=1}^{N} x_i^2$</td>
</tr>
<tr>
<td>Root-sum-of squares level</td>
<td>$T_{rss} = \sqrt{\sum_{i=1}^{N}</td>
</tr>
<tr>
<td>Mean Absolute Deviation</td>
<td>$T_{mad} = \frac{1}{N} \sum_{i=1}^{N}</td>
</tr>
</tbody>
</table>
### Table 3 Continued: Time Domain Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolute mean amplitude</td>
<td>$T_{ave} = \frac{1}{N} \sum_{i=1}^{N}</td>
</tr>
<tr>
<td>Shape Factor</td>
<td>$T_{sf} = \frac{T_{rms}}{T_{ave}}$</td>
</tr>
<tr>
<td>Peak</td>
<td>$T_{peak} = \max</td>
</tr>
<tr>
<td>Pulse factor</td>
<td>$T_{pf} = \frac{T_{max}}{T_m}$</td>
</tr>
<tr>
<td>Crest Factor</td>
<td>$T_{cf} = \frac{T_{peak}}{T_{rms}}$</td>
</tr>
<tr>
<td>Impulse factor</td>
<td>$T_{if} = \frac{T_{peak}}{T_{ave}}$</td>
</tr>
<tr>
<td>Clearance factor</td>
<td>$T_{clf} = \frac{T_{peak}}{T_{sra}}$</td>
</tr>
<tr>
<td>Margin factor</td>
<td>$T_{df} = \frac{T_{max}}{T_{sra}}$</td>
</tr>
<tr>
<td>Peak to Peak</td>
<td>$T_{p-p} = T_{max} - T_{min}$</td>
</tr>
</tbody>
</table>

Where $x_i$ is the time signal for $i=1,2,\ldots,N$. $N$ is the number of data points.

### Table 4: Frequency Domain Features

<table>
<thead>
<tr>
<th>Features and Expressions</th>
</tr>
</thead>
<tbody>
<tr>
<td>$FD_1 = \frac{\sum_{k=1}^{K} s(k)}{K}$</td>
</tr>
<tr>
<td>$FD_2 = \frac{\sum_{k=1}^{K} (s(k) - FD_1)^2}{K - 1}$</td>
</tr>
<tr>
<td>$FD_3 = \frac{\sum_{k=1}^{K} (s(k) - FD_1)^3}{K(\sqrt{FD_2})^3}$</td>
</tr>
<tr>
<td>$FD_4 = \frac{\sum_{k=1}^{K} (s(k) - FD_1)^4}{KFD_2^2}$</td>
</tr>
<tr>
<td>$FD_8 = \sqrt[4]{\frac{\sum_{k=1}^{K} f_k^4 s(k)}{\sum_{k=1}^{K} f_k^2 s(k)}}$</td>
</tr>
<tr>
<td>$FD_9 = \frac{\sum_{k=1}^{K} f_k^2 s(k)}{\sqrt{\sum_{k=1}^{K} s(k)} \sum_{k=1}^{K} f_k^4 s(k)}$</td>
</tr>
<tr>
<td>$FD_{10} = \frac{FD_6}{FD_5}$</td>
</tr>
<tr>
<td>$FD_{11} = \frac{\sum_{k=1}^{K} (f_k - FD_8)^3 s(k)}{KFD_6^3}$</td>
</tr>
</tbody>
</table>
Table 4 Continued: Frequency Domain Features

\[
FD_5 = \frac{\sum_{k=1}^{K} f_k s(k)}{s(k)}
\]

\[
FD_6 = \frac{\sum_{k=1}^{K} (f_k - FD_5)^2 s(k)}{K}
\]

\[
FD_7 = \sqrt{\frac{\sum_{k=1}^{K} f_k^2 s(k)}{\sum_{k=1}^{K} s(k)}}
\]

\[
FD_{12} = \frac{\sum_{k=1}^{K} (f_k - FD_5)^4 s(k)}{KFD_6^4}
\]

\[
FD_{13} = \frac{\sum_{k=1}^{K} (f_k - FD_5)^2 s(k)}{K\sqrt{FD_6}}
\]

Where \( s(k) \) is the amplitude of the frequency component for \( k = 1, 2, \ldots, K \). \( K \) is the number of spectrum lines. \( f_k \) is the frequency value of the \( k \)th spectrum line.

5.5 Feature Level Fusion Method

As shown in Figure 24, features from multiple sensors were merged into a single set and fed to the classifiers for machine condition monitoring system. Feature level fusion combines various features such as mean, median, crest factor, skewness, standard variance, kurtosis, etc. into a feature map that is used for segmentation and detection.

![Figure 24: Feature level fusion from [49].](image)

5.6 Feature Selection (FS)

In the previous step, a lot of possible features were extracted from 18 vibration and current signals and then, we combined these features to increase the diagnostic accuracy of our model as a result of enhanced complementary information. However, it is not guaranteed that all the 630 features contain useful information for our classification purpose and a lot of features may be correlated which is not desirable. Moreover, the increased number of features correspondingly increases the training time due to the high dimensionality of the feature set. FS is an effective method that can be used to select the most discriminative features and enhance the performance of the data-driven diagnostic system in terms of either accuracy or computational time [50]. In this study, the ensemble learning model, MRMR (Minimum Redundancy Maximum Relevance) method in MATLAB is used for performing FS. MRMR method represents a set of mutually exclusive and highly differentiated critical features and accurately reflects the response
variable. This algorithm allows for identifying essential features and redundant features using mutual information of variables.

After implementing the MRMR method on our data set, the most informative features are recognized based on their prediction ability and can be used to train the SVM-based classification scheme. To increase the generalization ability of the developed classifier and to avoid the curse of dimensionality, we have selected the top 30 features in Figure 25.

![Feature Rank](image)

**Figure 25: Feature Rank**

### 5.7 SVM Model

The function `fitcecoc` was used in the development of the SVM model in MATLAB, which involves setting a variety of parameters. The entire dataset was initially split into 70% training set and 30% test set, and the training set was used to train the SVM model. The RBF kernel was used for the kernel function. The C limits the penalty levied on large-residual findings. A higher restriction on the C offers a more robust model. A smaller value offers a stiffer model and less susceptible to over-fitting.
6 Results and Discussion

In Section 6.1, the accuracy of the binary classification problem using the current sensor is calculated under three operating conditions (full-load, full-load and half-load, full-load, half-load and no-load). In Section 6.2, the accuracy of different classes is calculated by using a single sensor in the multiclass classification system. In Section 6.3, feature level fusion technology is implemented and their results are compared with Section 6.2 results. Each fault was given a specific class as shown in Table 5.

Table 5: Fault classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Fault Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Dynamic eccentricity skewed rotor</td>
</tr>
<tr>
<td>2</td>
<td>Dynamic eccentricity two non-consecutive broken bars loose bearing skewed rotor</td>
</tr>
<tr>
<td>3</td>
<td>Healthy Skewed rotor</td>
</tr>
<tr>
<td>4</td>
<td>One broken bar skewed rotor</td>
</tr>
<tr>
<td>5</td>
<td>Three broken bar skewed rotor</td>
</tr>
<tr>
<td>6</td>
<td>Two broken bar skewed rotor</td>
</tr>
<tr>
<td>7</td>
<td>Two non-consecutive broken bars dynamic eccentricity skewed rotor</td>
</tr>
<tr>
<td>8</td>
<td>Two non-consecutive broken bars no eccentricity skewed rotor</td>
</tr>
</tbody>
</table>

The Table 6 shows the number of samples found at each operation condition in each class.

Table 6: Number of samples

<table>
<thead>
<tr>
<th>Operation Condition</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full-Load</td>
<td>290</td>
<td>513</td>
<td>146</td>
<td>158</td>
<td>82</td>
<td>201</td>
<td>291</td>
<td>333</td>
</tr>
<tr>
<td>Half-Load</td>
<td>290</td>
<td>330</td>
<td>125</td>
<td>157</td>
<td>62</td>
<td>117</td>
<td>297</td>
<td>288</td>
</tr>
<tr>
<td>No-Load</td>
<td>164</td>
<td>230</td>
<td>76</td>
<td>157</td>
<td>73</td>
<td>73</td>
<td>350</td>
<td>260</td>
</tr>
</tbody>
</table>

An equivalent numbers of samples from each class were used when increasing the classes as shown in Table 7. For example, in a full-load condition, the total number of samples were 290 in Class 01, 146 in Class 03, and 201 in Class 06. The lowest number of 146 common samples in each class was chosen by combining these.

A Combination of the operating condition was divided into three parts shown in Table 8. Condition 01 contains only full-load; Condition 2 is a combination of full and half-load, Condition 3 is a combination of full, half and no-load.
Table 7: Combined samples

<table>
<thead>
<tr>
<th>Classification Problems</th>
<th>3-Class</th>
<th>4-Class</th>
<th>5-Class</th>
<th>6-Class</th>
<th>7-Class</th>
<th>8-Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Combined Classes</td>
<td>1 3 6</td>
<td>1 3 4 6</td>
<td>1 3 4 6 7</td>
<td>1 3 4 6 7 8</td>
<td>1 3 4 5 6 7 8</td>
<td>1 2 3 4 5 6 7 8</td>
</tr>
<tr>
<td>LCS (Full-Load)</td>
<td>146</td>
<td>146</td>
<td>82</td>
<td>82</td>
<td>82</td>
<td>82</td>
</tr>
<tr>
<td>LCS (Half-Load)</td>
<td>117</td>
<td>117</td>
<td>62</td>
<td>62</td>
<td>62</td>
<td>62</td>
</tr>
<tr>
<td>LCS (No-Load)</td>
<td>73</td>
<td>73</td>
<td>73</td>
<td>73</td>
<td>73</td>
<td>73</td>
</tr>
</tbody>
</table>

LCS - Lowest number of common samples

Table 8: Operating Conditions

<table>
<thead>
<tr>
<th>Operating Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condition 1</td>
</tr>
<tr>
<td>Condition 2</td>
</tr>
<tr>
<td>Condition 3</td>
</tr>
</tbody>
</table>

6.1 Binary Classification

PCA vs FS

Initially, the 2-class problem was tested in all three operating conditions (F, FH, FHN) using a current sensor. However, PCA and FS approaches were also checked to minimize the features due to the large number of features contained in the dataset. In the first scenario of the implementation, the SVM model was tested with F condition, then gradually increased to FH and FHN. The results show that the classification accuracy of the model is 100% with PCA in F and FH conditions; however, the accuracy is significantly reduced in FHN condition. Figure 26 illustrates the 2-class problem, which PCA successfully characterizes in F and FH conditions. Likewise, the FS was tested for all three operating conditions. The results indicate FS in FHN is more accurate compared to PCA.

![Figure 26](image)

Figure 26: (a) PCA classification on operating condition 1, (b) PCA classification on operating condition 2
Similarly, with the increase in classes and operating conditions, better results are obtained from FS when testing PCA and FS. Accordingly, the FS method was used for the upcoming test.

6.2 Multiclass Classification

3 Class Problem (1 3 and 6)

SVM is generally a binary classifier, but it is built as a multiclass classifier for the purpose of this study. In all three operating conditions, it is evaluated for the 3-class problem.

1. Full-Load

![Confusion Matrix](image1)

![Confusion Matrix](image2)

Figure 27: (a) 3-Class full-load training confusion matrix (b) 3 Class full-load testing confusion matrix

2. Full and Half-Load

![Confusion Matrix](image3)

![Confusion Matrix](image4)

Figure 28: (a) 3 Class full and half-load training confusion matrix (b) 3 Class full and half-load testing confusion matrix
3. Full Half and No-Load

![Training confusion matrix](image1)

![Testing confusion matrix](image2)

Figure 29: (a) 3 Class full-load, half-load and no-load training confusion matrix (b) 3 Class full-load, half-load and no load testing confusion matrix

As shown in Figure 27 at F condition, both training and testing dataset provides 100% accuracy. However, a 98.5% accuracy in FH condition further drops to 97.1% in FHN condition. The results indicated that the accuracy decreases with increase in operating condition. Similarly, for the combination of all eight classes, experiments were carried out in three different combinations of operating conditions and the results are shown in Table 9. The accuracy of the SVM model is calculated by the following equation (38).

\[
\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}
\]  

(38)

<table>
<thead>
<tr>
<th>Classes</th>
<th>O/Conditions</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 Class</td>
<td>1 3 6</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>98.5</td>
</tr>
<tr>
<td></td>
<td>FH</td>
<td>97.1</td>
</tr>
<tr>
<td></td>
<td>FHN</td>
<td></td>
</tr>
<tr>
<td>4 Class</td>
<td>1 3 4 6</td>
<td>99.5</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>99.1</td>
</tr>
<tr>
<td></td>
<td>FH</td>
<td>97.2</td>
</tr>
<tr>
<td></td>
<td>FHN</td>
<td></td>
</tr>
<tr>
<td>5 Class</td>
<td>1 3 4 5 6</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>96.4</td>
</tr>
<tr>
<td></td>
<td>FH</td>
<td>96.3</td>
</tr>
<tr>
<td></td>
<td>FHN</td>
<td></td>
</tr>
<tr>
<td>6 Class</td>
<td>1 3 4 5 6 7</td>
<td>98.9</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>95.5</td>
</tr>
<tr>
<td></td>
<td>FH</td>
<td>94.4</td>
</tr>
<tr>
<td></td>
<td>FHN</td>
<td></td>
</tr>
<tr>
<td>7 Class</td>
<td>1 3 4 5 6 7 8</td>
<td>99.3</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>97.5</td>
</tr>
<tr>
<td></td>
<td>FH</td>
<td>93.9</td>
</tr>
<tr>
<td></td>
<td>FHN</td>
<td></td>
</tr>
<tr>
<td>8 Class</td>
<td>1 2 3 4 5 6 7 8</td>
<td>99.3</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>93.4</td>
</tr>
<tr>
<td></td>
<td>FH</td>
<td>92.4</td>
</tr>
<tr>
<td></td>
<td>FHN</td>
<td></td>
</tr>
</tbody>
</table>
Figure 30 shows the classification accuracy shown in Table 9 of the SVM model in six faulty combined classes of 3 Class, 4 Class, 5 Class, 6 Class, 7 Class and 8 Class. Overall, the F condition effectively preserves an accuracy of over 99% across all combined classes. Nevertheless, accuracy gradually decreases FH conditions. But when it combines with FHN condition, the accuracy of the Class 3 problem is 97.1%, and rapidly decreases to 92% in the 8 class problem.

![Figure 30: Faulty classes vs Accuracy](image)

8 Class (Full Half and No-Load)

![Figure 31: (a) 8 Class full-load half-load and no-load training confusion matrix (b) 8 Class full-load, half-load and no load testing confusion matrix](image)
Most of the samples are misclassified in the training and test confusion matrix, as seen in Figure 31 as a result of which, efficiency is reduced considerably. In order to improve the accuracy of the model, the feature level fusion method was used.

6.3 Sensor fusion method

This feature level fusion approach incorporates features from different sensors into one set. As shown in Figure 32 the 1216 samples in the training data are fully characterized when the model is trained using the feature-level fusion method. Similarly, when testing with 520 test samples, 9 samples could not be accurately classified. Notwithstanding the inaccuracy, 99.98% efficiency could still be obtained.

![Training confusion matrix](image1)

![Testing confusion matrix](image2)

Figure 32: (a) 8 Class full-load, half-load and no-load sensor fusion training confusion matrix (b) 8 Class full-load, half and no load sensor fusion testing confusion matrix

The *fitcecoc* classifier model in MATLAB chooses the best parameter among one-vs-one or one-vs-all for multiclass classification. As shown in Table 10 one-vs-one approach was used when experimenting with one sensor, while one-vs-all was used when testing with a sensor fusion method. As discussed in the previous chapter, the largest C controls the SVM model, and it does not allow any errors. Similarly, when C is small, the model allows for some error, thus achieving greater generalization. In our case, for both methods, there is no significant change in C. Since C has a small value; it allows some error and makes the separating hyper plane larger. However, when using one sensor, the kernel scale, which is 1.436, changes to 120.26 when using the multiple sensor fusion. This is a massive change. In conclusion, the key reason for the increase in sensor fusion efficiency is a major kernel-level improvement. Also, the use of multiple sensor fusion method reduces the evaluation time.
6.4 Accuracy versus Number of selected features

Seven and eight class problems were examined to see how the accuracy varied by increasing the number of features. As shown in Figure 33, the overall accuracy gradually increases with number of features, but at some point, accuracy can be seen to decrease. As such, this function is not monotonically increasing. This indicates that features are sometimes too complex or contain insufficient information.

![Image](7 Class classification No.of selected features vs Accuracy)

![Image](8 Class classification No.of selected features vs Accuracy)

Figure 33: (a) 7 Class - Accuracy versus Number of selected features Accuracy (b) 8 Class - Accuracy versus Number of selected features

6.5 Accuracy versus Number of fused sensors

This part evaluates how the accuracy changes with changes in the number of multiple fused sensors. For this analysis a total of eighteen sensors were used including current and vibration sensors. Each sensor has different types of information, and a particular sensor may not receive enough information. In spite of this, the sensor fusion technology was used, and by fusing the sensor we can observe the increase in accuracy. As shown in Figure 34, the three sensors are initially fused to the eight-class problem. The test was performed by increasing each of the three sensors. When tested with eighteen sensors, the accuracy increased upto 99.8%.
Figure 34: (a) 7 Class - Number of fused sensors (b) 8 Class - Accuracy versus Number of fused sensors
7 Conclusion

The thesis has developed a feature level fusion-based approach of condition monitoring of IM using SVM. Both current and vibration sensors are used for training the classifier. The proposed method consists of four components: data preprocessing, feature extraction, feature selection, designing and tuning the machine-learning model. Data preprocessing technique transforms the raw data into a useable and workable format. It involves data standardization, regulatory of incomplete data and deletion of irrelevant data. During the feature extraction stage thirty-five features are extracted from the time and frequency domain. During FS stage, the essential features with the most information are selected from the primary feature set. Finally, a multiclass SVM is designed by MATLAB `fitcecoc` function. The associated hyperparameters are adjusted during training.

The experiment has been conducted in three parts; binary classification, multiclass classification, and sensor fusion. In binary classification, the 2-class problem is tested, and the performance of the designed model is verified. The feature selection methods of PCA and MRMR have also been evaluated. The experimental results demonstrate that the appropriate feature set can be effectively selected by the MRMR method. The accuracy of different classes is checked by using a single sensor in a multiclass classification method. According to the experimental results, at the full-load condition, the accuracy is 100% for three-class classification problem. With increase in number of classes, the accuracy reduces but not significantly. When checked with FH and FHN conditions, the accuracy decreases dramatically to 92%. However, by implementing feature level fusion technology, accuracy can be improved to 99.8%. The results prove that the use of sensor fusion can achieve greater efficiency compared to using a single sensor. The experimental results indicate the effectiveness of the proposed method. Accuracy has been further analyzed by increasing the number of features and the number of fused sensors in the sensor fusion technique. Future works include focusing on the state of the art sensor fusion techniques and exploring their effect on the performance of the fault diagnosis system.
References


