# Using WiSARD classifier to evaluate the performance of various pre-trained networks for clutch fault diagnosis

Sairam Sridharan, V. Sugumaran, Naveen Venkatesh S

**ABSTRACT**

A clutch is an automotive component that plays a vital role in manual transmission vehicles. Its main function is to engage and disengage the connection between the engine's power and the transmission, which ultimately affects the wheels. Multiple signs of clutch system problems include difficulty in shifting gears, poor performance, slippage and potentially dangerous driving conditions. The fault diagnosis of the clutch is of significant importance, with the main focus being providing safe and effective vehicle operation. An experimental test rig operating under No-load conditions to obtain vibration signals that represented the state of the clutch. Vibration signals were acquired for 1 healthy condition and 5 different fault conditions, namely release fingers worn (RFW), pressure plate broken (PPB), tangential strip bent (TSB), pressure plate worn (PPW), friction material loss (FML) of the clutch. The vibration signals are captured using the uniaxial accelerometer. The vibration signals underwent scalogram image transform and the image datasets were resized. Feature extraction process was done using 5 pre-trained networks: AlexNet, GoogleNet, VGG16, ResNet-50, DenseNet-201. The extracted features were recorded and feature selection process was performed using J48 decision tree algorithm. The study involves the use of deep learning and machine learning for the fault diagnosis process. The test accuracies of each pre-trained network obtained after applying the WiSARD classifier was compared, with DenseNet-201 having the best accuracy of 95.83%. The best accuracy model was noted so as to provide a proper predictive maintenance model for the clutch system fault diagnosis.

**Keywords:** Clutch, Fault diagnosis, Scalogram, AlexNet, GoogleNet, VGG16, ResNet-50, DenseNet-201, WiSARD

## 1. Introduction

In automobiles, manual transmission is the most common transmission type. Some have automated manual transmission (AMT). The major difference is that AMT is equipped with smart actuator systems for power transmission [1]. The clutch is a crucial mechanical component in manual transmission vehicles. The clutch acts as an essential link between the engine and transmission, that facilitates gear changes and helps in regulating or controlling the transmission of power to the wheels. In a manual transmission vehicle, the clutch is located in between the gearbox and the engine of the powertrain. A clutch disc, pressure plate and release bearing are some important parts that collectively make up the clutch. When the clutch pedal is pressed by the driver, the clutch disengages, cutting the power link between the engine and the transmission. This disconnection or cutting allows the driver to change different gears or come to a complete stop without stalling the vehicle's engine. The clutch is gradually engaged once the clutch pedal is released, transmitting power from the engine to the wheels and enabling the car to move.

A properly functioning clutch ensures the safe operation of the vehicle and when the clutch system experiences any issues or malfunctions, it results in severe problems like slippage, engine noises, difficulty in gear shifting, reduced performance of the vehicle. These are the reasons to have a health clutch system in a vehicle. The importance of fault diagnosis of clutch cannot be overstated, addressing these problems lowers the risk of damage to the transmission system and mishaps while driving. Clutch fault diagnosis allows early detecting, repairing and replacing defective components, thereby helping to be cost-effective. In general, clutch fault detection is crucial for reliable, cost-efficient and safe vehicle operation. Automated data driven approach to fault diagnosis have been used to combine signal processing, statistics and machine learning methods to continuously monitor and diagnose the target component [2]. In many engineering applications such as the automotive industry, predictive maintenance and fault diagnosis have become increasingly crucial.

Real-time measurements are taken during the fault diagnosis process to assess the condition of the mechanical system, in this case, the clutch. These fault diagnosis methods are mainly classified into signal-based, knowledge-based and model-based approach. The model-based approach offers the benefit of being extremely precise, enabling an extensive in-depth analysis of the system complexities and faults from a mechanistic and structural perspective. However, it has certain limitations because of its complexity in real-world problems, which makes it difficult to develop precise mathematical models and limits its usefulness. The traditional fault diagnosis approach involves data acquisition, feature engineering and feature classification [3]. The data acquisition process involves signal acquisition from sensors incorporated in the mechanical systems. The most commonly obtained signals include temperature [4], current, speed, vibration and acoustic [5]. Vibrational signals are widely preferred for mechanical systems as they exhibit more intrinsic characteristics of the current condition of the system. It also accounts for precision and fault detection patterns [6], [7], [8]. The features are then extracted from these raw signals, the most significant features are then chosen and placed in the order of their importance. The significant features are selected by dimensionality reduction process such as decision tree and principal component analysis [9]. Finally, the features are classified using several classifiers. In this case, it is the WiSARD classifier, which is a Weightless neural network approach [10]. The accuracy and precision of results heavily rely on the level and quality of the output. It is still a challenging task to derive a complete understanding of the system and its faults from limited datasets. It is wise to choose the best course of action for such set of demands and requirements.

For increasing the dependability and accuracies, several researchers have scrutinized a variety of approaches ranging from traditional models to the recent Deep [11], CNN [12] and machine learning models in the field of fault diagnosis. The features acquired from the obtained signals have a significant impact on the performance of the fault diagnosis model. To determine the best possible approach, a high level of domain expertise and knowledge are required. Deep learning has made it feasible for extremely large datasets to be easily classified. For instance, the research paper [13] have utilized deep convolutional neural networks to classify 1.2 million high-resolution images into 1000 different classes.

Deep learning models with multiple layers are able to learn and process more complex features leading to improved performances in classification. The successful application using 16-19 weight layers helped facilitate enhanced classification [14]. Adaptive weighted multiscale convolutional neural networks (AWMSCNN) were used for fault diagnosis in [15], the AWMSCNN achieved the best performance and converged faster than all other models. The paper [16] proposed fault diagnosis for switchgear using ultrasound analysis and extreme learning machine (ELM) which is a subset of artificial neural networks (ANN).

Fault diagnosis of automotive systems, especially the clutch is a vital aspect in vehicle repair and maintenance. The main focus is to attain high accuracy in the diagnosis, it is possible by deploying machine learning and deep learning methods to create models for issues in the real-world. This paper presents a comprehensive methodology that includes several essential aspects. It illustrates the experimental setup for the test, elucidates the feature extraction by using scalogram image type derived features as they detain the important features, feature selection using J48 decision tree classifier to highlight the important features in the dataset, so that model can achieve high accuracy by only selecting the relevant features for classification [17]. This study explores the application of weightless neural networks (WiSARD) for feature classification. The WiSARD classifier is a RAM-based neural network model [18], these are weightless neural networks basically used for classification and pattern recognition [19]. It has been used for several different applications such as sorting based on image recognition [20], data stream clustering [21], robot prosthetic control and navigation [22], forecasting of time series [23] and much more.

By using proper live setup, the vibrational signals were collected in No load condition for 5 different fault conditions and 1 healthy condition. The fault conditions were Release Fingers Worn (RFW), Pressure Plate Broken (PPB), Tangential Strip Bent (TSB), Pressure Plate Worn (PPW), Friction Material Loss (FML) (Refer Figure-2). In this paper, the performance of numerous modern pre-trained networks, including AlexNet, GoogleNet, VGG16, ResNet-50 and DenseNet-201, is being evaluated in the context of fault diagnosis of different conditions in the clutch from the image dataset derived from the extracted vibration signals. Experiments with these pre-trained networks were carried out, and the results were tabulated and compared to establish the best network for fault diagnosis of clutch systems.

The major contributions of this paper can be summarized in the following way:

- Feature extraction using pre-trained networks (Deep Learning)
- Feature selection using J48 decision tree algorithm
- Feature Classification using WiSARD classifier

**Table 1 - SUMMARY OF RELATED WORKS**

| REFERENCE | COMPONENT | TECHNIQUES USED |
|---|---|---|
| [24] | Centrifugal Pumps | • Continuous Wavelet Transform (CWT) to breakdown the vibrational signals<br>• Scalogram based imaging<br>• Adaptive Deep Convolutional Neural Network (ADCNN) |
| [25] | Dynamic systems (univariate and multi- variate) | • Fault detection and diagnosis based on weightless neural networks (FDD-WiSARD)<br>• Using moving average methods for time series mapping of the data<br>• Use of RAM devices for neurons in weightless neural networks |
| [26] | Centrifugal Pump | • Using CNN for fault classification<br>• Using Sobel filter for edge extraction<br>• Training the CNN on SobelEdge scalograms to acquire the features |
| [18] | Photovoltaic Modules | • Weightless neural network (WiSARD classifier)<br>• Image transform<br>• Utilization of pre-trained network – DenseNet-201 |
| [27] | Automotive clutch system | • Machine learning based approach on vibration analysis for condition monitoring of automotive clutch.<br>• Feature selection by J48 decision tree algorithm<br>• Feature classification using Bayes based classifiers |
| [28] | Friction Disc of Dry Clutch | • Finite-element analysis to determine temperature<br>• Using MATLAB for obtaining solutions to the finite-element model |

**Research gaps**

- It is critical to develop more advanced approaches for early detection and diagnosis of clutch faults. Traditional methods detect the problems only when it becomes serious. The application of machine learning and deep learning models helps in providing early, accurate and efficient detection and diagnosis of these faults.
- Vibrational signals are proved to be better in fault diagnosis and condition monitoring because of its high sensitivity and its ability to provide quantitative information about the fault.
- Most of the studies ignored filtering of the signal to cut down steps, which have resulted in slight inaccuracies. But, the application of scalogram transform has helped in reducing noises and enhancing the feature extraction process and aids in accurate fault diagnosis.
- WiSARD networks useful in data analysis and pattern recognition because of their memory. Better accuracies have been obtained in models using WiSARD classifiers.

**Novelty**

The study's novelty revolves around the use of weightless neural networks (WiSARD) classifier) in the fault diagnosis of six different clutch system conditions, namely good, release fingers worn, pressure plate broken, tangential strip bent, pressure plate worn, friction material loss. The vibration signals of each were taken and the dataset was augmented using image transforms to enhance the acquired data. A combination of deep learning and machine learning was utilized in this study. The features were extracted for five pre-trained networks namely AlexNet, GoogleNet, VGG16, ResNet-50 and DenseNet-201. Feature selection was carried out with the help of J48 decision tree algorithm. The selected features were pruned using J48 once again to have a smaller number of features with those contributing to the highest accuracy. The selected features were trained using the WiSARD classifier and the accuracy for each pre-trained network was noted down. The accuracies for each network were optimised by tuning the hyperparameters, such as bit number, bleach confidence, bleach flag, bleach step, map type and tic number. The results of the study showed the effectiveness of the proposed methodology and the WiSARD classifier's ability to provide great accuracies in classification.

The rest of the paper is arranged in the following order: Section 2 describes the experimental studies, Section 3 explains the methodology involved, Section 4 holds the results and discussion and about the classifiers used, Section 5 is conclusion

## 2. Experimental studies

This section on experimental research tells us how the vibrational data was acquired from the laboratory test setup.

### 2.1 Experimental setup

The experiment was carried out on a single plate dry friction clutch from an automobile that was available (Maruti Zen). The setup included an accelerometer, an AC motor, a clutch assembly, a flywheel and four bearings. A DAQ card was installed to acquire the vibrational signals [3]. Refer to Figure-1 for the live-setup. The AC motor was operated at a rotational speed of 1400 rpm. A shaft 200 mm in length and 25 mm in diameter was used to link the flywheel to the motor, and a clutch assembly was attached to shaft. The experimental setup was completed with multiple bearings that supported the shaft. A uniaxial accelerometer was used to measure the vibrations in the system. The accelerometer had a sensitivity of 10.26 mV/g. The accelerometer was mounted on top of the bearing and the vibrations were measured. The other end of the accelerometer was connected to a DAQ system, in which the vibration signals were converted from analog to digital. The experiment was conducted in a No-load condition.

**2.2 Data acquisition**

Once the accelerometer took the readings, the output was sent to the NI 9234 DAQ card through an USB connector. The vibration signals were then processed and converted from analog to digital signals with the help of an Analog-to-Digital converter (ADC). The plots of the vibration signals were made from the digital values. The same process was carried out for the 5 fault conditions and 1 healthy condition of the clutch. Image transform, scalogram was carried out to enhance the results. The augmented data was further processed and resized to 224x224 and 227x227 for the pre-trained networks. The parameters considered during signal collection are mentioned below:

- Sampling frequency – 25 kHz
- Sampling length – 8192 steps
- Number of samples for each condition – 100
- Load condition – No Load

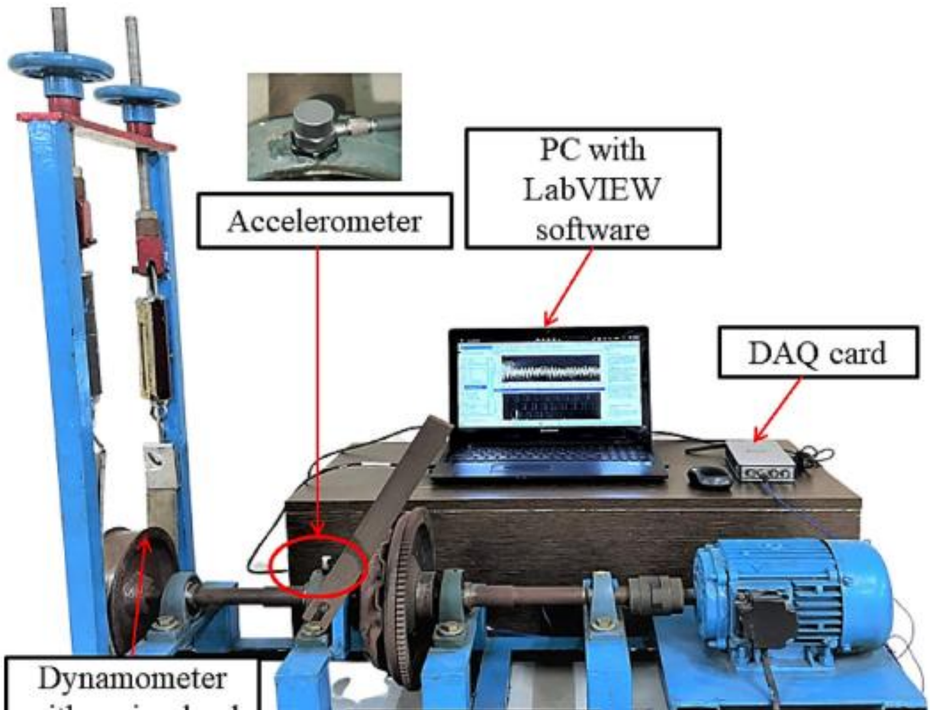


**Figure 1 - Experimental setup**

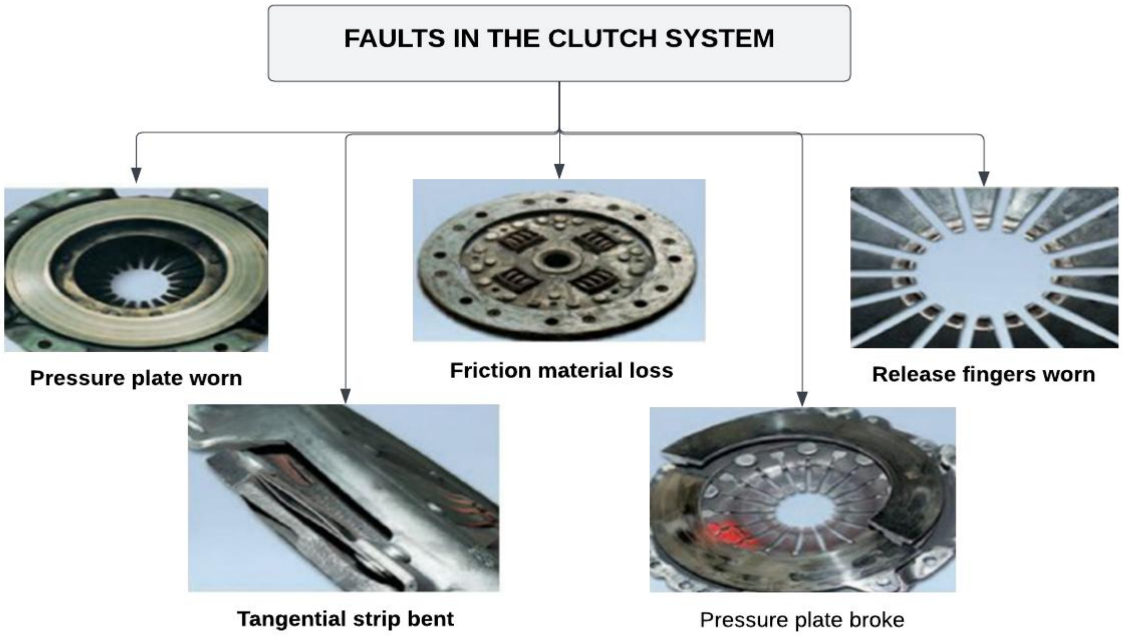


**Figure 2 - Various faults in the clutch system**

## 3. Methodology

The fault diagnosis methodology for the clutch system included four stages:

i. Resizing the image dataset
ii. Feature extraction: Extraction of features using pre-trained networks such as AlexNet, GoogleNet, VGG16, ResNet-50 and DenseNet-201
iii. Feature selection using J48 decision tree
iv. Feature classification using WiSARD classifier

### 3.1 Resizing and pre-processing

Firstly, the acquired vibration signals are stored in the form of scalogram type images (vibration plots). Scalogram is a signal processing method used for precise analysis and reduce noise from the acquired signals. It makes the analysis more effective in detecting faults. The image dataset is for 5 fault conditions (FML, PPB, PPW, RFW, TSB) and 1 good condition. These images are pre-processed and resized to 227x227 for AlexNet and 224x224 for GoogleNet, VGG 16, ResNet-50 and DenseNet-201 (Refer Figure-3). These are the several renowned pre-trained networks used to classify the image dataset and detect the condition of the clutch system.

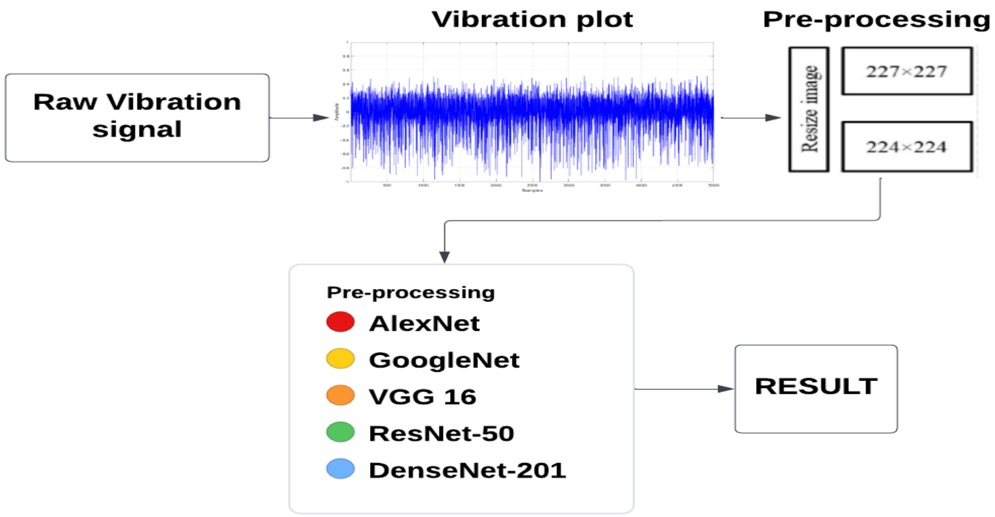


**Figure 3 - workflow using pre-trained networks**

### 3.1.1 AlexNet pre-trained network

AlexNet, which was developed in 2012 by Alex Krizhevsky, is one of the pioneering deep convolutional neural networks that helped in a revolution in deep learning. This network went through training on an exclusively large dataset comprising over 1.2 million images, with 1000 distinct image classes. AlexNet emerged as a top performer in the annual ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Despite its compact architecture, AlexNet has 8 layers and a staggering 61 million learnable parameters. The network model was specifically designed for images with an input size of 227x227 pixels. Each layer of the network employs a rectified linear unit (ReLU) as its activation function, to facilitate the resolution of non-linear problems.

### 3.1.2 GoogleNet pre-trained network

GoogleNet was developed in 2014 by a group of researchers at Google. It is also known as the inception architecture. The GoogleNet has a deep and complex architecture with multiple parallel convolutional layers. This architecture, which consists of 22 layers, has been used to a variety of fields, including facial recognition, robotics, and adversarial training. These models excel in identifying complex features by utilizing convolutional layers with varying filter sizes. This method significantly lowers the dimensionality of the data while minimizing the computational overhead.

### 3.1.3 VGG 16 pre-trained network

VGG 16 was developed by Karen Simonyan and Andrew Zisserman belonging to the Visual Geometry Group (VGG) in the year 2014. It secured the title of best performing network in the ILSVRC competition. VGG 16's architectural frame consists of 13 convolutional layers, 3 fully connected layers, 5 max pooling layers and a classification layer. The VGG 16 architecture's initial layers are dedicated to learning the fundamental features such as edges, while the deeper layers are assigned with extracting and understanding more complex feature information. Each image is resized and deconstructed in memory during the convolutional process to uncover more relevant and significant features.

#### 3.1.4 ResNet-50 pre-trained network

ResNet (Residual Network) was developed by Kaiming He et al. in 2015. It was the top performing model in 2015 ILSVRC competition. It is used primarily because of its rapid convergence and precise classification capabilities. The ResNet architecture exists in several variation. In this study we focus on ResNet-50. The ResNet-50 architecture consists of 49 convolutional layers and a single fully connected layer. The Residual network is eight times deeper than VGG 16 and can encompass very large number of learnable features.

#### 3.1.5 DenseNet-201 pre-trained network

Densely connected convolutional network (DenseNet) is a neural network architecture developed in 2017. It has the ability to leverage dense connections between layers, resulting in improved gradient flow and feature reuse. DenseNet has a wide range on application like image classification, image generation and object detection. In DenseNet, each layer receives input from the preceding layers and sends the output to all the subsequent layers. The dense connectivity helps in feature reuse. Transition blocks aid in controlling growth of feature maps and reduce the computational complexity. The DenseNet-201 is a densely connected convolutional neural network with very efficient memory usage. It has a total of 201 layers and 20 million learnable parameters. The network size is 80 MB and accepts image input size of 224x224 pixels.

### 3.2 Feature Extraction

The feature extraction process minimizes the number of variable parameters needed to interpret and evaluate a larger dataset. In this study, the pre-trained networks were used to extract features from the clutch system. 1000 features for all the conditions were extracted using each pre-trained networks. The derived features were saved as ".csv" files for further processing. The convolutional neural networks (CNN) creates these features that automatically differentiate between each class based on the given labels. Transfer learning has developed into an effective approach for extracting and categorizing unique image data by making only a few to the last few layers.

### 3.3 Feature Selection

The feature selection process involves identifying and selecting the most relevant features that can be used in classification and fault diagnosis. The irrelevant features may reduce the performance of the classifier and increase the processing complexity. The feature selection technique eliminates the irrelevant features and improve the classification model performance [29]. Decision tree algorithms are often used due to their effective information retrieval capabilities. J48 is a well-known decision tree algorithm used for feature selection process [17]. The J48 features were collected and reviewed again to remove the least significant features and have an appropriate number of relevant features. This process is called decision tree pruning. The features in lowest nodes generally contribute very less and are removed to ease the computation. The datasets were then split into Training (80%) and Testing (20%) sets.

**Table 2 - Selected features**

| PRE-TRAINED NETWORKS | NUMBER OF FEATURES AFTER PRUNING | SELECTED FEATURES AFTER PRUNING |
|---|---|---|
| **AlexNet** | **26 Features** | feature_304, feature_138, feature_883, feature_231, feature_693, feature_160, feature_409, feature_31, feature_420, feature_256, feature_5, feature_812, feature_863, feature_927, feature_2, feature_803, feature_12, feature_758, feature_145, feature_381, feature_898, feature_125, feature_223, feature_18, feature_97, feature_924 |
| **GoogleNet** | **31 Features** | feature_929, feature_648, feature_533, feature_374, feature_146, feature_162, feature_292, feature_2, feature_125, feature_105, feature_36, feature_239, feature_684, feature_771, feature_3, feature_361, feature_505, feature_905, feature_385, feature_395, feature_822, feature_119, feature_52, feature_100, feature_71, feature_382, feature_164, feature_579, feature_195, feature_59, feature_857 |
| **VGG 16** | **28 Features** | feature_973, feature_680, feature_824, feature_531, feature_734, feature_44, feature_899, feature_321, feature_602, feature_921, feature_3, feature_572, feature_331, feature_437, feature_4, feature_740, feature_166, feature_923, feature_675, feature_570, feature_11, feature_829, feature_828, feature_207, feature_319, feature_27, feature_178, feature_595 |
| **ResNet-50** | **19 Features** | feature_430, feature_355, feature_467, feature_933, feature_283, feature_531, feature_592, feature_85, feature_624, feature_868, feature_24, feature_607, feature_399, feature_826, feature_1, feature_858, feature_2, feature_158, feature_415 |
| **DenseNet-201** | **34 Features** | feature_771, feature_255, feature_389, feature_361, feature_703, feature_615, feature_182, feature_168, feature_6, feature_705, feature_489, feature_393, feature_441, feature_873, feature_896, feature_339, feature_451, feature_1, feature_586, feature_236, feature_4, feature_199, feature_461, feature_693, feature_249, feature_264, feature_9, feature_532, feature_300, feature_294, feature_736, feature_33, feature_384, feature_867 |

### 3.4 Feature Classification

**WiSARD classifier**

The feature classification is performed using WiSARD classifier. WiSARD is a weightless neural model that stores the function computed by each neuron in look up tables rather than in the weights of a neuron connection. The major advantage of this classifier is its simplicity and execution speed. The training of the WiSARD classifier is a one-shot memorization process. It relatively simple and straightforward than other classifiers.

Weightless neural networks (WNN) are facilitated by RAM devices that store significant image features as address (binary initialization). Weighted neural networks, on the other hand, apply weights to comprehensive collection of features (Refer Figure-4). In terms of neural network representation computation and real-world scenarios, WNN outperform weighted neural networks. Weightless neural networks use memory locations and hashing [30], while weighted neural networks used weighted synapses for the adjustment of signal strength between neurons.


**Figure 4 - Difference between weightless and weighted neural networks**

The output from RAM consists of either 0 or 1. The composition of RAM devices in WNN can aid in applying several architectures, the most popular of which being WiSARD. The WiSARD structure is made up of two or more discriminators, which correspond to the number of user-defined classes. Each discriminator is allocated a certain class that is trained independently to learn all the patterns allotted to the class. The discriminator module consists of RAM devices which map these input patterns. The RAM devices study and learn sub-patterns from a modelled map of an input pattern, which can be represented as a tuple. When the testing cycle is completed, the RAM device identifies the memory address of each related tuple and returns the stored value are 0 or 1. Figure-5 represents the outline of the WiSARD classifier.

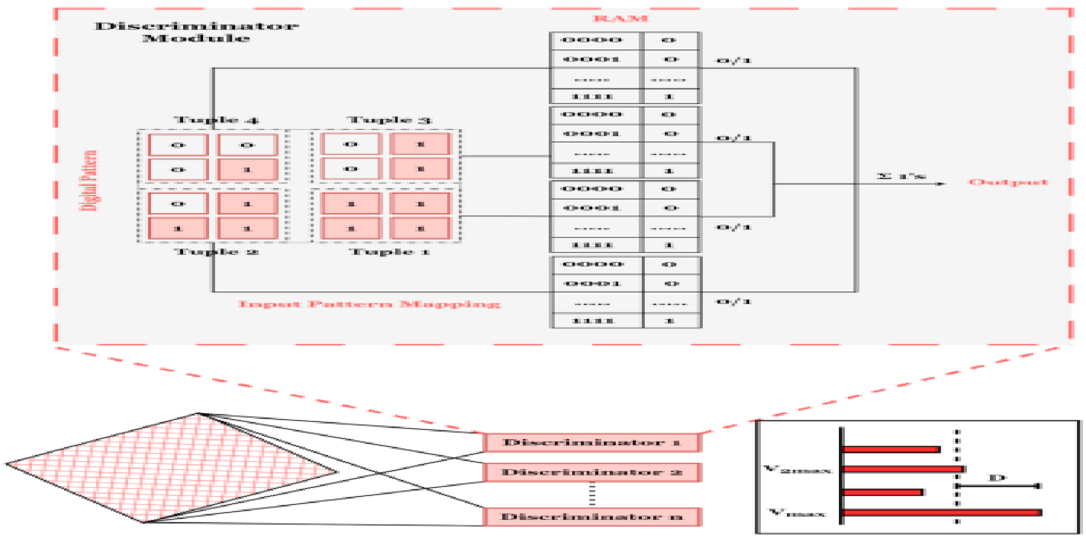


**Figure 5 - WiSARD classifier with discriminator module**

The 5 pre-trained networks were classified using the WiSARD classifier once the feature selection process was done. Training accuracy, validation accuracy and testing accuracy were all noted down. The network which had the best classification accuracy was termed the best for the fault diagnosis process.

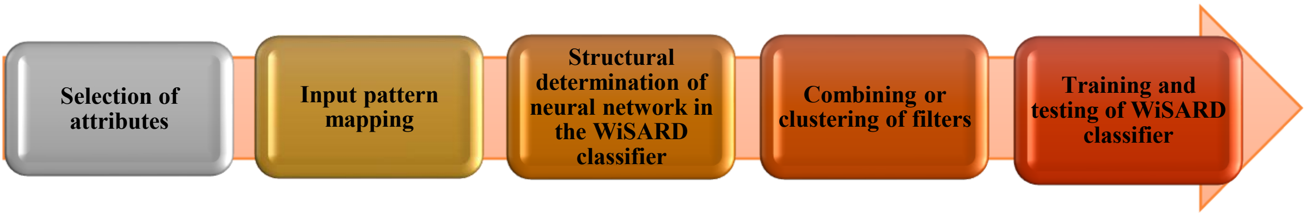


**Figure 6 - Overall working of the WiSARD classifier**

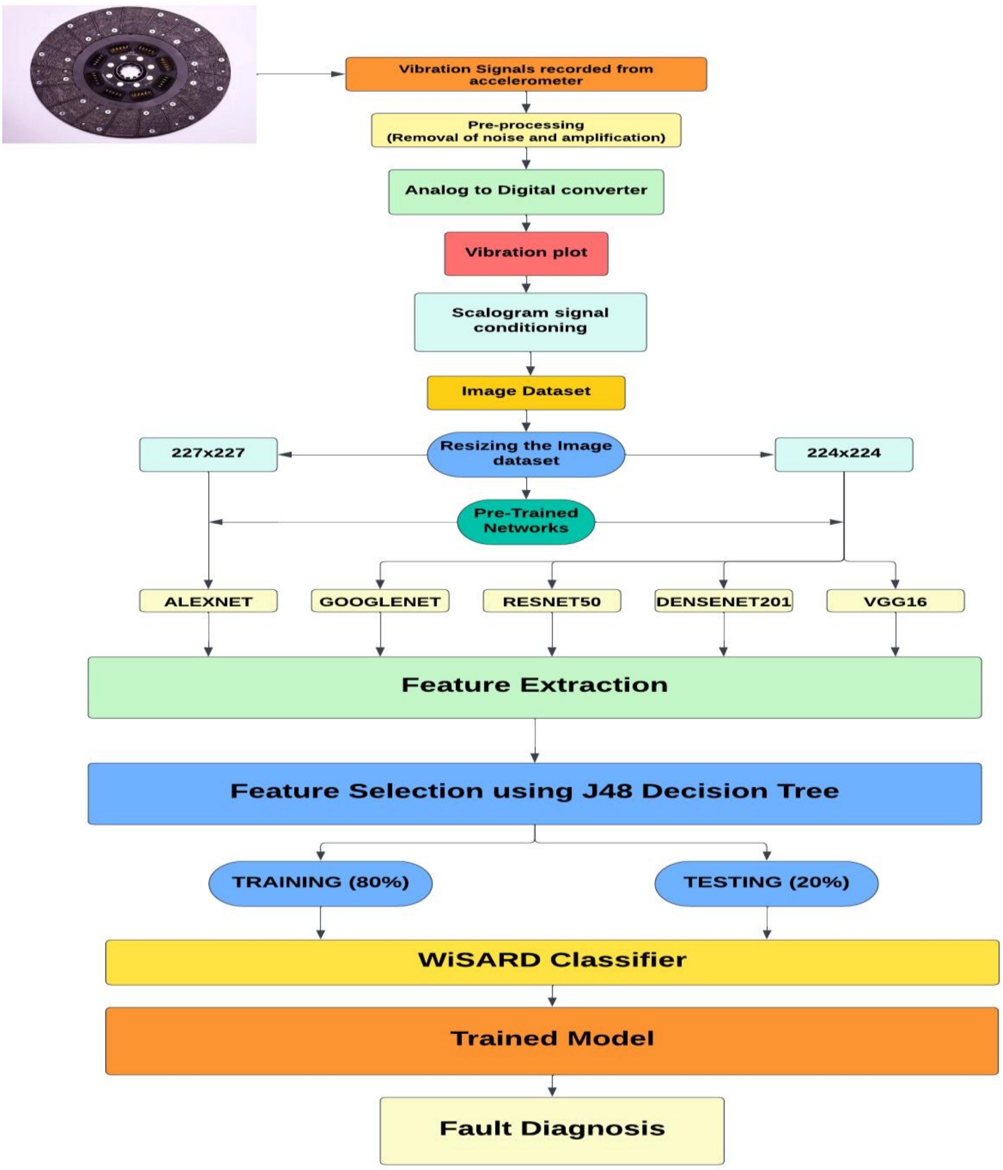


**Figure 7 - Proposed methodology**

## 4. Results and Discussion

This study aimed to evaluate the performance of the WiSARD classifier in the fault detection and diagnosis of a clutch system under No-load condition. The performance of the classifier was evaluated for 1 healthy condition and 5 fault conditions, namely Release Fingers Worn (RFW), Pressure Plate Broken (PPB), Tangential Strip Bent (TSB), Pressure Plate Worn (PPW), Friction Material Loss (FML). The various hyperparameters of the WiSARD classifier such as bit number, bleach confidence, bleach flag, bleach step, map type and tic number were changed accordingly to arrive at an optimal combination, in order to achieve the best accuracy. The dataset was split into Training set (80%) and Testing set (20%) while validation accuracy was found out using ten-fold cross validation. The feature extraction from the pre-trained networks were carried out in MATLAB using the deep learning toolbox. Feature selection and classification was performed using Weka.

### 4.1 Impact of changing Bit Number

The number of bits used to represent the data in sparse distribution representation (SDR) has an impact on how well the WiSARD classifier performs. Bits are individual binary units that state some information. The randomized and initialized binary bits are used by the WiSARD classifier for pattern recognition. Different bit numbers were used for this classification, including 2, 4, 8, 16, 32, rest all the hyperparameters were unchanged. The bit number that provided the best test accuracy for each pre-trained network are mentioned below in Table-3.

**Table 3 - Determining the best bit number for each network**

| Pre-trained networks | Bit No | Training accuracy (%) | Cross validation (%) | Test accuracy (%) |
|---|---|---|---|---|
| AlexNet | 32 | 100.00 | 82.29 | 83.33 |
| GoogleNet | 16 | 100.00 | 88.75 | 85.83 |
| VGG 16 | 16 | 100.00 | 88.75 | 89.17 |
| ResNet-50 | 32 | 100.00 | 93.13 | 91.67 |
| DenseNet-201 | 16 | 100.00 | 92.29 | 95.83 |

### 4.2 Impact of changing Bleach Confidence

The bleach confidence parameter in the WiSARD classifier helps in adjusting the prediction confidence, by considering the similarity between the incoming and stored patterns. The confidence increases when the input matches the pattern, otherwise it decreases. This aids in improving the predictability of the classifier. Different bleach confidence values, including 0.6, 0.7, 0.8, 0.9 and 1, were used in this experiment. The bit number which gave the best accuracy was set for determining the best bleach confidence for each network. The rest hyperparameters were unchanged. The bleach confidence that provided the best test accuracy for each pre-trained network are mentioned below in Table-4.

**Table 4 - Determining the best bleach confidence for each network**

| Pre-trained networks | Bleach Confidence | Training accuracy (%) | Cross validation (%) | Test accuracy (%) |
|---|---|---|---|---|
| AlexNet | 0.7 | 100.00 | 82.29 | 85.00 |
| GoogleNet | 0.7 | 100.00 | 88.75 | 85.83 |
| VGG 16 | 0.7 | 100.00 | 87.92 | 90.00 |
| ResNet-50 | 0.6 | 100.00 | 92.71 | 93.33 |
| DenseNet-201 | 0.9 | 100.00 | 92.29 | 95.83 |

### 4.3 Impact of changing Bleach Flag

The bleach flag of the WiSARD classifier behaves more like a switch. The prediction confidence changes when the bleach flag is set to true. When set to false, the prediction confidence is unaffected. The flag regulates whether and how the classifier modifies the confidence when making a judgement call. The two settings of the bleach flag are: True or False. The bit number and bleach confidence which gave the best accuracy was set for determining the best bleach flag setting for each network. The rest hyperparameters were unchanged. The bleach flag that provided the best test accuracy for each pre-trained network is mentioned below in Table-5.

**Table 5 - Determining the best bleach flag for each network**

| Pre-trained networks | Bleach Flag | Training accuracy (%) | Cross validation (%) | Test accuracy (%) |
|---|---|---|---|---|
| AlexNet | FALSE | 100.00 | 82.29 | 85.00 |
| GoogleNet | FALSE | 100.00 | 88.75 | 85.83 |
| VGG 16 | FALSE | 100.00 | 87.92 | 90.00 |
| ResNet-50 | FALSE | 100.00 | 93.13 | 93.33 |
| DenseNet-201 | FALSE | 100.00 | 92.92 | 95.83 |

### 4.4 Impact of changing Bleach Step

The bleach step is used to prevent overfitting, which occurs when a model becomes overly focused on training data. The WiSARD classifier's bleach step hyperparameter helps in regulating the bleaching speed. Low bleach step value slows it down and higher value speeds it up. In order to get the best test accuracy, the experiment used the values 1,2,5 and 10 as bleach step for various iterations. During this process, the bit number, bleach confidence and bleach flag that gave the best accuracy was set for determining the best bleach step for each network. The rest hyperparameters were unchanged. The bleach step that provided the best test accuracy for each pre-trained network are mentioned below in Table-6.

**Table 6 - Determining the best bleach step for each network**

| Pre-trained networks | Bleach Step | Training accuracy (%) | Cross validation (%) | Test accuracy (%) |
|---|---|---|---|---|
| AlexNet | 1 | 100.00 | 83.75 | 85.00 |
| GoogleNet | 1 | 100.00 | 88.75 | 85.83 |
| VGG 16 | 1 | 100.00 | 87.92 | 90.00 |
| ResNet-50 | 1 | 100.00 | 93.13 | 93.33 |
| DenseNet-201 | 1 | 100.00 | 92.92 | 95.83 |

### 4.5 Impact of changing Map Type

Map type represents the way of connecting input patterns with the bit cells in the classifier. Linear and random map types are mostly used by the WiSARD classifier. In linear map, the input patterns are systematically assigned to particular bits. In random map, the input patterns are randomly linked to bit cells in a non-sequential manner. The bit number, bleach confidence, bleach flag and bleach step that gave the best accuracy was set for determining the best map type for each network. The rest hyperparameters were unchanged. The map type that provided the best test accuracy for each pre-trained network are mentioned below in Table-7.

**Table 7 - Determining the best map type for each network**

| Pre-trained networks | MAP TYPE | Training accuracy (%) | Cross validation (%) | Test accuracy (%) |
|---|---|---|---|---|
| AlexNet | RANDOM | 100.00 | 82.91 | 85.00 |
| GoogleNet | RANDOM | 100.00 | 88.75 | 85.83 |
| VGG 16 | RANDOM | 100.00 | 87.92 | 90.00 |
| ResNet-50 | RANDOM | 100.00 | 93.13 | 93.33 |
| DenseNet-201 | RANDOM | 100.00 | 92.92 | 95.83 |

### 4.6 Impact of changing Tic Number

The tic number denotes the bits inside each cell that influence the memory capacity and resolutions. More tics provide detailed memories, but they also use more memory and increase the computation load. In order to improve classifier performance, it requires balancing memory and efficiency, which is often optimized through experimentation. The various tic number values like 1, 10, 20, 50, 100 and 256 were used for the experiment. The bit number, bleach confidence, bleach flag, bleach step and map type that gave the best accuracy was set for determining the best tic number for each network. The rest hyperparameters were unchanged. The tic number that provided the best test accuracy for each pre-trained network are mentioned below in Table-8.

**Table 8 - Determining the best tic number for each network**

| Pre-trained networks | Tic No | Training accuracy (%) | Cross validation (%) | Test accuracy (%) |
|---|---|---|---|---|
| AlexNet | 256 | 100.00 | 82.71 | 85.00 |
| GoogleNet | 256 | 100.00 | 88.75 | 85.83 |
| VGG 16 | 100 | 100.00 | 87.92 | 90.83 |
| ResNet-50 | 256 | 100.00 | 93.13 | 93.33 |
| DenseNet-201 | 256 | 100.00 | 92.92 | 95.83 |

### 4.6 Best Hyperparameters

Table below presents the optimal hyperparameter selection for each pre-trained network based on the experiments carried out above. Out of all the networks, DenseNet-201 showed the best testing accuracy of 95.83 %. This study emphasizes the significance of implementing proper hyperparameters in the WiSARD classifier. Furthermore, the study successfully illustrated the efficacy of the proposed fault diagnosis process using the WiSARD classifier.

**Table 9 - Optimal hyperparameters of the WiSARD classifier for each network**

| Pre-trained networks | Bit No | Bleach Confidence | Bleach Flag | Bleach Step | Map Type | Tic No | Test set accuracy (%) |
|---|---|---|---|---|---|---|---|
| AlexNet | 32 | 0.7 | FALSE | 1 | RANDOM | 256 | 85 |
| GoogleNet | 16 | 0.7 | FALSE | 1 | RANDOM | 256 | 85.83 |
| VGG 16 | 16 | 0.7 | FALSE | 1 | RANDOM | 100 | 90.83 |
| ResNet-50 | 32 | 0.6 | FALSE | 1 | RANDOM | 256 | 93.33 |
| DenseNet-201 | 16 | 0.9 | FALSE | 1 | RANDOM | 256 | 95.83 |

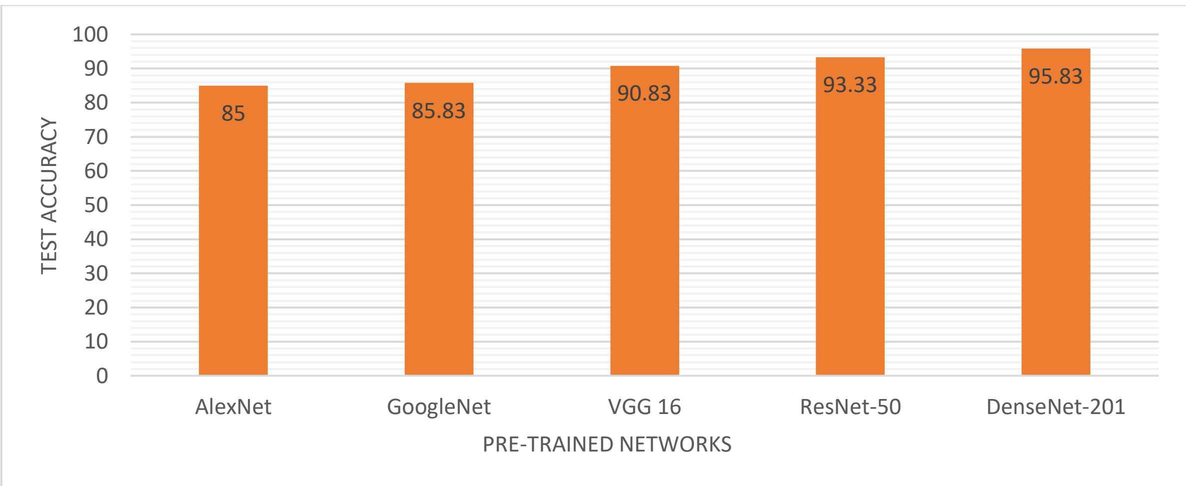


**Figure 8 - Comparing the Test accuracy of each network**

| TARGET / OUTPUT | FML | GOOD | PPB | PPW | RFW | TSB | SUM |
|---|---|---|---|---|---|---|---|
| FML | 20<br>16.67% | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 20<br>100.00%<br>0.00% |
| GOOD | 0<br>0.00% | 20<br>16.67% | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 20<br>100.00%<br>0.00% |
| PPB | 0<br>0.00% | 0<br>0.00% | 20<br>16.67% | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 20<br>100.00%<br>0.00% |
| PPW | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 18<br>15.00% | 1<br>0.83% | 1<br>0.83% | 20<br>90.00%<br>10.00% |
| RFW | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 1<br>0.83% | 19<br>15.83% | 0<br>0.00% | 20<br>95.00%<br>5.00% |
| TSB | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 0<br>0.00% | 2<br>1.67% | 18<br>15.00% | 20<br>90.00%<br>10.00% |
| SUM | 20<br>100.00%<br>0.00% | 20<br>100.00%<br>0.00% | 20<br>100.00%<br>0.00% | 19<br>94.74%<br>5.26% | 22<br>86.36%<br>13.64% | 19<br>94.74%<br>5.26% | 115 / 120<br>95.83%<br>4.17% |

**Figure 9 - Confusion matrix of WiSARD classifier with optimal hyperparameters for DenseNet-201**

## 5 Conclusion

The present study used a weightless neural network (WiSARD classifier) to classify several clutch system conditions. An experimental rig was setup to help with the study and vibration signals were measured for six different clutch conditions. The experiment was carried out under no load conditions and included five faulty and one healthy condition. The fault conditions considered were release fingers worn (RFW), pressure plate broken (PPB), tangential strip bent (TSB), pressure plate worn (PPW), friction material loss (FML). The effectiveness of deep learning and machine learning algorithms in fault diagnosis of a mechanical system was examined using a hybrid approach involving feature extraction using pre-trained networks (deep learning), namely AlexNet, GoogleNet, VGG16, ResNet-50, DenseNet-201, followed by feature selection using J48 decision tree algorithm and classification using WiSARD classifier. The dataset was divided into training (80%) and testing (20%) subsets. DenseNet-201 stood out among the other network models for its excellent accuracy of 95.83% when classified using the WiSARD classifier. As a result, among the numerous models considered in this study, DenseNet-201 performed best, making it a good contender for real-time clutch fault diagnosis problems. The proposed work can aid in accurate and quick fault diagnosis that can help enhance the vehicle performance, gear shifting maneuvers and most importantly the safety. This approach can offer solutions for timely fault detection and diagnosis of the clutch system. However, there are certain limitations in this context, such as costly sensors. To address such challenges, optimal and low-cost sensors can be used that can provide with proper signals.

## REFERENCES:


[1] A. Della Gatta, L. Iannelli, M. Pisaturo, A. Senatore, and F. Vasca, “A survey on modeling and engagement control for automotive dry clutch,” *Mechatronics*, vol. 55. Elsevier Ltd, pp. 63–75, Nov. 01, 2018. doi: 10.1016/j.mechatronics.2018.08.002.

[2] T. Olsson *et al.*, “Fault Diagnosis of Heavy Duty Machines: Automatic Transmission Clutches.”

[3] G. Chakrapani and V. Sugumaran, “Transfer learning based fault diagnosis of automobile dry clutch system,” *Eng Appl Artif Intell*, vol. 117, Jan. 2023, doi: 10.1016/j.engappai.2022.105522.

[4] A. M. D. Younus and B. S. Yang, “Intelligent fault diagnosis of rotating machinery using infrared thermal image,” in *Expert Systems with Applications*, Feb. 2012, pp. 2082–2091. doi: 10.1016/j.eswa.2011.08.004.

[5] Z. Liu, X. Wang, and L. Zhang, “Fault Diagnosis of Industrial Wind Turbine Blade Bearing Using Acoustic Emission Analysis,” *IEEE Trans Instrum Meas*, vol. 69, no. 9, pp. 6630–6639, Sep. 2020, doi: 10.1109/TIM.2020.2969062.

[6] M. Vishwakarma, R. Purohit, V. Harshlata, and P. Rajput, “Vibration Analysis & Condition Monitoring for Rotating Machines: A Review,” in *Materials Today: Proceedings*, Elsevier Ltd, 2017, pp. 2659–2664. doi: 10.1016/j.matpr.2017.02.140.

[7] F. Al-Badour, M. Sunar, and L. Cheded, “Vibration analysis of rotating machinery using time-frequency analysis and wavelet techniques,” *Mech Syst Signal Process*, vol. 25, no. 6, pp. 2083–2101, Aug. 2011, doi: 10.1016/j.ymssp.2011.01.017.

[8] A. Joshuva and V. Sugumaran, “A lazy learning approach for condition monitoring of wind turbine blade using vibration signals and histogram features,” *Measurement (Lond)*, vol. 152, Feb. 2020, doi: 10.1016/j.measurement.2019.107295.

[9] A. Mehta, D. Goyal, A. Choudhary, B. S. Pabla, and S. Belghith, “Machine Learning-Based Fault Diagnosis of Self-Aligning Bearings for Rotating Machinery Using Infrared Thermography,” *Math Probl Eng*, vol. 2021, 2021, doi: 10.1155/2021/9947300.

[10] ESANN 24. Brügge, M. Verleysen, Université catholique de Louvain, Katholieke Universiteit Leuven, C. I. and M. L. 24 2016. 04. 27-29 B. European Symposium on Artificial Neural Networks, and ESANN 24 2016.04.27-29 Bruges, *24th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning ESANN 2016 : Bruges, Belgium, April 27-28-29, 2016 : proceedings*.

[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Computer Society, Dec. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[12] M. Liang, P. Cao, and J. Tang, “Rolling bearing fault diagnosis based on feature fusion with parallel convolutional neural network,” *International Journal of Advanced Manufacturing Technology*, vol. 112, no. 3–4, pp. 819–831, Jan. 2021, doi: 10.1007/s00170-020-06401-8.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks.” [Online]. Available: http://code.google.com/p/cuda-convnet/

[14] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” Sep. 2014, [Online]. Available: http://arxiv.org/abs/1409.1556

[15] H. Qiao, T. Wang, P. Wang, L. Zhang, and M. Xu, “An Adaptive Weighted Multiscale Convolutional Neural Network for Rotating Machinery Fault Diagnosis under Variable Operating Conditions,” *IEEE Access*, vol. 7, pp. 118954–118964, 2019, doi: 10.1109/ACCESS.2019.2936625.

[16] S. Ishak, C. T. Yaw, S. P. Koh, S. K. Tiong, C. P. Chen, and T. Yusaf, “Fault classification system for switchgear cbm from an ultrasound analysis technique using extreme learning machine,” *Energies (Basel)*, vol. 14, no. 19, Oct. 2021, doi: 10.3390/en14196279.

[17] R. Ade, “Feature Selection based Classification using Naive Bayes, J48 and Support Vector Machine Dipali Bhosale,” 2014.

[18] N. V. Sridharan, J. V. Joseph, S. Vaithiyanathan, and M. Aghaei, “Weightless Neural Network-Based Detection and Diagnosis of Visual Faults in Photovoltaic Modules,” *Energies (Basel)*, vol. 16, no. 15, Aug. 2023, doi: 10.3390/en16155824.

[19] A. S. L. Filho, G. P. Guarisa, L. A. D. L. Filho, L. F. R. Oliveira, F. M. G. Franca, and P. M. V. Lima, “wisardpkg -- A library for WiSARD-based models,” May 2020, [Online]. Available: http://arxiv.org/abs/2005.00887

[20] I. Aleksander, W. V. Thomas, and P. A. Bowden, “Wisard a radical step forward in image recognition,” *Sensor Review*, vol. 4, no. 3. pp. 120-rere, Mar. 01, 1984. doi: 10.1108/eb007637.

[21] D. Cardoso, M. De Gregorio, P. Lima, ao Gama, and F. França, “A Weightless Neural Network-Based Approach for Stream Data Clustering.”

[22] M. Staffa, M. Giordano, and F. Ficuciello, “A WiSARD Network Approach for a BCI-Based Robotic Prosthetic Control,” *Int J Soc Robot*, vol. 12, no. 3, pp. 749–764, Jul. 2020, doi: 10.1007/s12369-019-00576-1.

[23] M. Dixon, IEEE Computer Society, and Institute of Electrical and Electronics Engineers, *Proceedings of the 3rd Workshop on High Performance Computational Finance : 2010, New Orleans, LA, USA : WHPCF ’10*.

[24] M. J. Hasan, A. Rai, Z. Ahmad, and J. M. Kim, “A Fault Diagnosis Framework for Centrifugal Pumps by Scalogram-Based Imaging and Deep Learning,” *IEEE Access*, vol. 9, pp. 58052–58066, 2021, doi: 10.1109/ACCESS.2021.3072854.

[25] J. C. M. Oliveira, K. V. Pontes, I. Sartori, and M. Embiruçu, “Fault Detection and Diagnosis in dynamic systems using Weightless Neural Networks,” *Expert Syst Appl*, vol. 84, pp. 200–219, Oct. 2017, doi: 10.1016/j.eswa.2017.05.020.

[26] W. Zaman, Z. Ahmad, M. F. Siddique, N. Ullah, and J. M. Kim, “Centrifugal Pump Fault Diagnosis Based on a Novel SobelEdge Scalogram and CNN,” *Sensors*, vol. 23, no. 11, Jun. 2023, doi: 10.3390/s23115255.

[27] G. Chakrapani and V. Sugumaran, “Health monitoring of automotive clutch system by using Bayes algorithms,” *IOP Conf Ser Mater Sci Eng*, vol. 1012, no. 1, p. 012028, Jan. 2021, doi: 10.1088/1757-899x/1012/1/012028.

[28] F. Meng and J. Xi, “Numerical and experimental investigation of temperature distribution for dry-clutches,” *Machines*, vol. 9, no. 9, Sep. 2021, doi: 10.3390/machines9090185.

[29] V. Sugumaran and K. I. Ramachandran, “Effect of number of features on classification of roller bearing faults using SVM and PSVM,” *Expert Syst Appl*, vol. 38, no. 4, pp. 4088–4096, Apr. 2011, doi: 10.1016/j.eswa.2010.09.072.

[30] L. D. Seneviratne, P. Visuwan, and K. Althoefer, “WEIGHTLESS NEURAL NETWORK BASED MONITORING OF SCREW FASTENINGS.”