DEVELOPMENT OF ANOMALY DETECTOR FOR MOTOR BEARING CONDITION MONITORING USING FAST FOURIER TRANSFORM (FFT) AND LONG SHORT TERM MEMORY (LSTM)-AUTOENCODER

By

SULAIMON A. BASHIR *

OLADEBO SULIAT JIMOH **

IDRIS MOHAMMED KOLO ***

ENESI FEMI AMINU ****

*-**** *Department of Computer Science, School of Information and Communication Technology, Federal University of Technology, Minna, Nigeria.*

Date Received: 16/05/2023 *Date Revised: 25/05/2023* *Date Accepted: 05/06/2023*

ABSTRACT

Anomaly detection in motor bearings is a critical task for preventing downtime and ensuring efficient operation. This paper proposes a novel approach for anomaly detection using Fast Fourier Transform (FFT) and Long Short-Term Memory (LSTM)-Autoencoder (AE). A data processing approach based on FFT was developed to pre-process the raw sensor data. This helped to reduce noise and improve the Signal-to-Noise Ratio (SNR). Additionally, an anomaly detection model based on LSTM-Autoencoder was developed and trained on the pre-processed data. The proposed approach was able to detect anomalies at a low threshold and achieved a high accuracy score.

Keywords: Motor Bearing, Anomaly Detection, Deep Learning, Fast Fourier Transform, Long Short Term Memory, Autoencoder.

INTRODUCTION

In recent times, Artificial Intelligence (AI) has been employed in the detection of anomalies in machines. Some techniques used include the Support Vector Machine (SVM) (Lee et al., 2020), Bayesian Classifier, Neural Networks, and Deep Learning (DL) methods (Jiang et al., 2019). However, the imbalanced nature of the dataset generated from sensors poses challenges in accurate anomaly detection.

According to Chalapathy and Chawla (2019), Artificial Intelligence (AI), including Machine Learning and Deep Learning, is preferred over traditional anomaly detection methods because it can detect anomalies using data that may not be available in traditional methods. This allows for earlier detection of anomalies, which can help to prevent breakdowns.

This paper has objectives related to SDGs

Among AI approaches, Deep Learning (DL) has been shown to outperform traditional Machine Learning because of its flexibility in learning and representing data in a hierarchical form in a Neural Network. This allows for better performance in anomaly detection.

As a result, Deep Learning has been used in a variety of applications, including, financial fraud detection (Adewumi & Akinyelu, 2017), Intrusion Detection in cyberspace (Kwon et al., 2019), Big data anomaly detection in the Internet of Things (IoT) (Mohammadi et al., 2018), and Medical Image Analysis (Litjens et al., 2017).

According to Lee et al. (2018a, 2018b), the use of deep learning for anomaly detection has been increasing because of its ability to receive signal data and automatically pre-process it before extracting the features needed for anomaly detection.

Generally, deep learning techniques for anomaly detection require huge amounts of data. It is important to note that the data generated from sensors is often characterized by electrical noise and a low Signal-to-Noise ratio (SNR). This noise can make it difficult for deep learning models to accurately detect anomalies (Ahmad et al., 2020).

Recently, a lot of researchers have done extensive work on anomaly detection. Researchers have observed that large thresholds are often used in anomaly detection systems that use sensor data. This is because sensor data is often characterized by noise and low Signal-to-Noise Ratio (SNR), regardless of the technique used. This low SNR can often lead to false alerts or no alerts of anomalies, and the inability of deep learning models to detect weak signals.

The generation of weak signals by some sensors can make it difficult to detect anomalies. This is because the threshold for detecting anomalies is often set based on the error margin of normalcy. If the threshold is set too high, it may not be possible to detect weak signals, even if they are outside the range of normalcy.

Another challenge in detecting anomalies is the use of time-stamped data. Time-stamped data can be large and difficult to process. This can make it difficult to identify anomalies in a timely manner.

To improve the detection of anomalies, it is important to clean up the data using Digital Signal Processing techniques such as Fast Fourier Transform (FFT) (Jabczyński & Szczesniak, 1995; Vo et al., 2017). This will allow anomalies to be detected at lower thresholds, which indicates improved sensitivity. FFT is a robust technique for data characterized by low Signal-to-Noise Ratio (SNR), and it helps to convert sequence data from the time domain to the frequency domain. This makes feature extraction easier, even for data with low SNR. Additionally, using FFT can aid anomaly detection faster by reducing computational time and acting as a filter to clean up the data.

Therefore, this research proposes the use of Fast Fourier Transform and LSTM-Autoencoder for the faster, more sensitive, and more accurate detection of anomalies in motor bearings.

1. Related Work

Khadersab and Shivakumar (2018) presented how FFT and Inverse Fast Fourier Transform (IFFT) were used to analyze bearing failure. They compared healthy vibration data and faulty vibration data using these techniques to accurately assess the bearing failure. In their experiment, a piezoelectric accelerometer sensor was used to collect vibration data. The acquired data was connected to an FFT algorithm called EL-Calc. The FFT signal was then used to generate IFFT and a spectrogram. At the end of the experiment, defects were effectively identified.

According to Sulka et al. (2019), the causes of unwanted vibration and the extent of a fault can be estimated. They used FFT and Short-Time Fourier Transform (STFT) to identify defects in bearings. The difference between these two techniques is that FFT does not use a time window, while STFT uses a time window to analyze a particular vibration signal. After using both techniques, it was observed that FFT shows a larger amplitude in the vibration signal that was analyzed. The amplitude also depends on the severity of the damage.

Abouelanouar et al. (2018) used Wavelet Transform (WT) to detect faults in gears and bearings. They found that WT was a powerful tool for fault detection, even with non-periodic vibrational signals (Hruntovich et al., 2019).

Kanwal et al. (2019) presented a method for detecting tampering in images that used a combination of the Fast Fourier Transform (FFT), local texture descriptors, and a Support Vector Machine (SVM) classifier. The results of their research showed that this method was able to increase the accuracy of tampering detection.

Barot and Kulkarni (2021) reviewed different techniques for detecting anomalies in machines. The research emphasized the need for Digital Signal Processing (DSP) techniques, such as the Fast Fourier Transform (FFT), Wavelet Transform (WT), Discrete Wavelet Transform (DWT), and Wavelet Packet Transform (WPT). They also stressed the importance of denoising and extracting weak signals, which can be done using wavelet-based signal denoising. It also reviewed Artificial Intelligence (AI) methods for diagnosing faults in bearings.

To reduce downtime in industries, Jin et al. (2019) proposed a data-driven approach for bearing prognostics based on the Kolmogorov-Smirnov test, self-organizing map, and unscented Kalman filter, as shown in Figure 1. The first step in their approach is to detect bearing degradation, also known as anomaly detection, by learning from historical data generated by vibration sensors. The second step is to predict the Remaining Useful Life (RUL) of the bearing using a degradation model and an unscented Kalman filter.

Zhang et al. (2019) emphasized the use of the Internet of Things (IoT) for data gathering for detecting anomalies in bearings. This research work aims to ensure adequate maintenance before a complete machine breakdown, exposes different techniques used to track the degradation of the system. These techniques involve the use of signal processing for better performance in degradation tracking. Cyclic Spectral Correlation (CSC) and Cyclic Spectral Coherence (CSCoh) have been proven to be powerful tools for signal processing. Due to the difficulty of obtaining labeled data from the experimental setup in Figure 2, fault detection was performed using semi-supervised learning and Support Data Description (SDD).

Wang et al. (2019) presented a case study that utilized Conditional Based Maintenance (CBM) for centrifugal pumps as part of a safety program for critical water systems used to mitigate fire hazards. This was done to monitor the operational condition of centrifugal pumps. Vibration data was gathered to investigate if CBM could identify different faults in the centrifugal pump.

Figure 3 shows the workflow of the research, which involved determining the best practices or approaches for CBM in the lab compared to the CBM approach in industry.

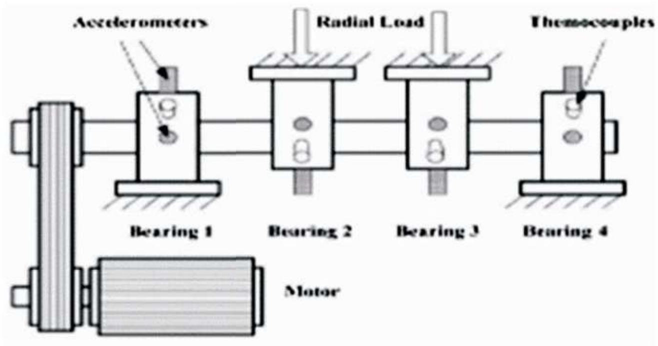

Figure 2. Experimental Set Up for Data Gathering

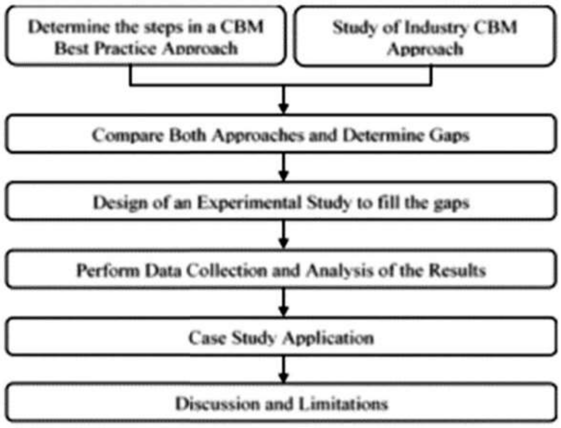

Figure 3. Work Flow of the Research

The advent of smart meters and the Internet of Things (IoT) has improved the gathering of data, which is instrumental to the detection of anomalies in industrial machines. According to Pittino et al. (2020), Machine Learning has been used to extract information from datasets, the important aspect of anomaly detection is that it aids in the detection of bearing faults. This helps to detect malfunctioning bearings before the machines completely break down.

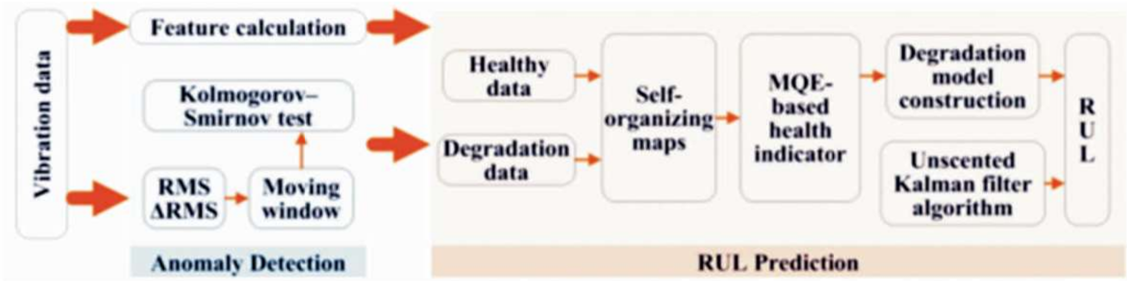

Figure 1. Data Driven Approach for Bearing Prognostic

Saeki et al. (2019) used visualization techniques, specifically Convolutional Neural Networks (CNNs), for the detection of motor bearing anomalies. In this work, a vibration dataset was captured and predictions were made to state the health status of the bearings. These predictions were compared with the analysis done by experts. Results showed that the technique is helpful in providing useful information about the health status of motor bearings.

Schmidt and Heyns (2019) state that anomaly detection can be used to identify localized faults in gears, even when historical fault data is not available. Discrepancy Analysis, Continuous Wavelet Transform (CWT), and Principal Component Analysis (PCA) are used to determine the divergence of the gear. Bayesian Data Analysis is then used to infer the presence of a localized anomaly. By identifying localized faults early, preventive maintenance can be performed to extend the life of the machine and avoid costly downtime.

Asakura et al. (2020) used Mahalanobis–Taguchi System (MTS), a method for quantifying the extent of damage in logical systems, to detect anomalies in those systems. They applied the technique to a large-scale vertical transfer system and developed calculations to achieve proper values for the technique based on simple excitation using a shaker.

Cooper et al. (2020) worked on anomaly detection in milling tools using Generative Adversarial Networks (GAN) and acoustic data. It uses K-Nearest Neighbors (KNN) to detect anomalies in machineries.

Nath (2020) disclosed that abrupt changes in sequential data are a major problem in anomaly detection. To solve this problem, low latency anomaly detection based on Quick Change Detection (QCD) is needed for effective detection. QCD minimizes the delay in detecting anomalies observed in sequential data. This is useful because in most models, the post-change distribution model may not be available. However, QCD has been used for bearing fault detection in turbines.

Sohaib and Kim (2018) proposed a method for detecting faults in bearings using bi-spectrum analysis and Convolutional Neural Networks (CNNs). First, the bi-spectrum of the vibrational signal was extracted. Next, a CNN based on a stochastic optimization function was used to extract the interclass of the bi-spectra. The proposed method was able to detect faults more accurately than the previous work.

Pandarakone et al. (2018) proposed an online bearing fault detection method based on deep learning. Fast Fourier transform (FFT) was used to perform spectral analysis on data generated from the load current of the stator coil. This helped to extract features from the data, which were then used to train a convolutional neural network (CNN). The CNN was able to classify different types of bearing faults, including single scratch (SS) and full scratch (FS). The average accuracy of the system was 88.17%. Abid et al. (2019) presented a technique to aid the detection of faults in bearings and the extent of the damage.

The technique Optimized Stationary Wavelet Packet Transform (OP-SWPT), is an advanced form of the Digital Signal Processing Technique Wavelet Transform. The authors used Fast Fourier Transform (FFT) to analyze the signal in the frequency domain, Short Time Fourier Transform (STFT) is used to identify the location of the extracted features, and Wavelet Transform (WT) for non-stationary signals. The results showed that the technique was effective in detecting different types of bearing faults. It was also reported that the technique detected faults faster than other methods.

Egaji et al. (2020) emphasized the need to use available data, usually vibration data, to avoid downtime in industries. It suggests that Digital Signal Processing methods such as FFT could be used to improve the output of the data, which is often noisy due to sensor inefficiency.

In Boniol et al. (2020), features were extracted from the data. These features were then used to train a Neural Network. To make detection easier, Principal Component Analysis (PCA) was used to reduce the dimension of the data from 24 to 1-dimensional space. The output from PCA was then used as input to a regression model, which reconstructs the input. The error between the input and the reconstructed output reveals anomaly detection. The regression models used in this research were Support Vector Machine (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN).

Industrial fans are used to aid cooling and ensure the proper functioning of industrial machines. Gong et al. (2018) presented an online solution to detect anomalies in industrial fans. To achieve this, they used acoustic signals with an intelligent prediction integrated system that is connected to the internet. Furthermore, Glowacz (2019) used Acoustic Signal Enhancement Filter and Adaptive Kalman Filter for feature extraction and detection.

Sohaib and Kim (2019) presented a method for detecting anomalies in machine bearings using complex envelope spectra and a Stacked Sparse Autoencoder-based Deep Neural Network (DNN) (Liu & Gryllias, 2020). To overcome the fluctuations in shaft speed, a fault diagnostic scheme was developed. The use of complex envelope spectra made the detection process easier.

The literature review shows that researchers have used either Digital Signal Processing (DSP) or Machine Learning (ML) to improve motor bearing detection (Vo et al., 2017, Wei et al., 2019). However, no studies have been found that use both the Fast Fourier transform (FFT) and Long Short-Term Memory (LSTM)-Autoencoders. Additionally, no other researchers have improved the performance of LSTM-Autoencoders for detecting motor bearing anomalies.

2. Methodology

The proposed technique in this research includes the use of National Aeronautics and Space Administration (NASA) data on motor bearings, the use of the Fast Fourier Transform (FFT) and a Long Short-Term Memory (LSTM)-Autoencoder to achieve anomaly detection. Python was used as a tool to write the code. The workflow of the proposed system is shown in Figure 4.

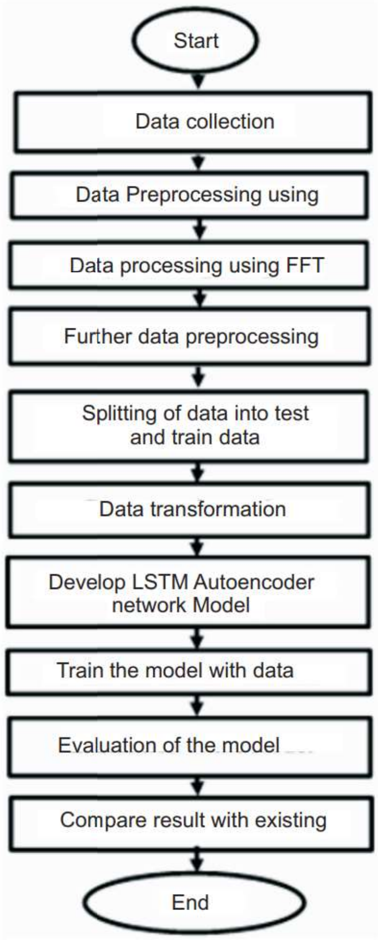

Figure 4. Proposed Model Diagram

In this research, the data was collected, pre-processed with pandas, and then processed with FFT to aid anomaly detection. The structured data characterized by normalcy was then fed into the proposed model and trained. Afterwards, the proposed model was tested with abnormal data and the performance of the model was then evaluated using two metrics threshold, also known as anomaly score, and accuracy.

2.1 Data Collection

The data used in this study is sensor readings obtained from vibration sensors fastened to motor bearings by NASA Acoustics. These data were generated from four bearings that were run until failure under constant load. The data, which was taken at 10-minute intervals contains 20,800 data points per bearing read at a sampling rate of 20 kHz.

Figure 5 shows the part of the data collected in files named with time and date stamp. Figure 6 shows the unorganized data before processing.

2.2 Data Pre-Processing

The vibration data for the four motor bearings was

☐	2004.02.12.10.52.39
☐	2004.02.12.11.02.39
☐	2004.02.12.11.12.39
☐	2004.02.12.11.22.39
☐	2004.02.12.11.32.39
☐	2004.02.12.11.42.39

Figure 5. Raw Data Collected with Date and Time Stamp

1	0.112	-0.010	-0.005	0.022
2	0.049	-0.012	-0.046	0.020
3	-0.027	-0.017	-0.012	0.098
4	-0.110	0.029	0.112	0.056
5	0.007	-0.024	-0.044	0.054
6	0.122	0.051	-0.007	-0.012
7	0.134	0.007	0.034	-0.017
8	-0.015	-0.027	0.002	0.027
9	-0.168	-0.037	-0.107	0.042
10	-0.061	-0.090	0.020	0.046
11	0.012	-0.059	0.100	0.012
12	0.027	0.061	-0.068	-0.032
13	-0.056	0.002	-0.081	0.022
14	-0.115	-0.049	-0.044	0.056
15	0.015	0.042	-0.046	0.005
16	0.100	0.049	-0.125	-0.020
17	0.088	0.076	0.020	-0.015

Figure 6. Unorganized Data Before Processing

collected. The data was tagged with dates of collection and stored in separate files. Pandas was used to merge the four files into a single dataset as shown in Figure 7. The dataset was inspected to verify that there were no missing data. The data type and shape were checked to ensure that the data conformed to the format required by the deep learning algorithm.

2.3 Data Pre-Processing using FFT

Fast Fourier Transform (FFT) is a technique that employs the transformation of time sequence signals to its frequency domain so as to extract latent behavior of the signal source. In other words, the FFT decomposes an N-point time domain signal into N-time domain signals, each composed of a single point. Afterwards, the FFT algorithm calculates the N-frequency spectra corresponding to the N-time domain signals. Lastly, the N-spectra are then synthesized into a single frequency spectrum. This algorithm utilizes Equation 1.

$$X_k = \sum_{n=0}^{N-1} x_n e^{\frac{i2\pi kn}{N}} \tag{1}$$

	Bearing 1	Bearing 2	Bearing 3	Bearing 4
2004-02-12 10:52:39	0.060236	0.074227	0.083926	0.044443
2004-02-12 11:02:39	0.061455	0.073844	0.084457	0.045081
2004-02-12 11:12:39	0.061361	0.075609	0.082837	0.045118
2004-02-12 11:22:39	0.061665	0.073279	0.084879	0.044172
2004-02-12 11:32:39	0.061944	0.074593	0.082626	0.044659

Figure 7. Dataset of the Motor Bearing After Pre-Processing

where N is the size of the domain, and $X_k=X_0,X_1......X_{N-1}$ is converted to another sequence number $x_n=x_0,x_1......x_{N-1}$ is the signal. In other words, x_n is a sinusoid with frequency of k/n, which is a cross correlation sequence of X_k. Furthermore, it must be noted that the 'N' points must be in the form of 2^n. This means the N-time points must be within this range, otherwise it would not be able to capture the whole data. During the decomposition, the levels of decomposition is given as $\log_2 N$. For example, a 16-point signal, also represented as (2^4), is broken down into x stages where $x = \log_2 24$. This results in a breakdown into 4 stages. During the breakdown, the original samples are re-ordered via bit reversal. Table 1 shows a sample data re-ordering.

The pseudocode for the Fast Fourier Transform is as follows,

//For general case let the input G of any case have a sequence

//G=($a_0, a_1, ...a_{N-1}$)

//Note that N is a power of 2. Also, we want to return output values of H //given as:

$$// \ H = A(x) = \sum_{j=0}^{N-1} a_j x^j$$

//where H is a polynomial similar to Equation 1, evaluated at Nth root of //unity and a is the coefficient of the polynomial.

if N=1 then return (a_0)

if N>1 then //calling the Fourier transforms recursively.

$(s_0, s_1, ..., s_{n/2-1}) = FFT((a_0, a_2, ..., a_{N-2})w^2)$ *// this deals with even sequence*

$(s'_0, s'_1, ..., s'_{n/2-1}) = FFT((a_1, a_3, ..., a_{N-1})w^2)$ *// this deals with an odd sequence.*

for j=0 to $\frac{N}{2}-1$

$$r_j = s_j + w_N{}^j s'_j$$

$$r_{j+\frac{N}{2}} = s_j - w_N{}^j s'_j$$ *// the negative sign is from the odd.*

// Note that w is the primitive Nth root of unity if $w^0, \ldots, w^{N-1}$ *//are root of the unity*

return $(r_0, r_1, \ldots, r_{N-1})$

end for

end if

end if

//How long will it take to do this computation?

$$T(N) = 2T(N/2) + O(N)$$

//N is the size of the problem and O(N) is the order of N as solved in the //Equation 1.

// $= O(N \log_N)$

//which is much better if the system is to run for $O(N^2)$.

2.4 Further Data Pre-Processing

Further pre-processing step was done to normalize the data. The normalization process ensures that all data points are converted to numbers between 0 and 1. LSTM models typically use data in a three-dimensional tensor format, which is important to reshape the data from its current two-dimensional format (date-time stamp, feature) to a three-dimensional format (date-sample, time-sample, features). Figure 8 shows the shape of the dataset that was originally collected.

Normal Sample	Binary Representation	Decomposed Outcome	Binary Representation
0	0000	0	0000
1	0001	8	1000
2	0010	4	0100
3	0011	12	1100
4	0100	2	0010
5	0101	10	1010
6	0110	6	0110
7	0111	14	1110
8	1000	1	0001
9	1001	9	1001
10	1010	5	0101
11	1011	13	1101
12	1100	3	0011
13	1101	11	1011
14	1110	7	0111
15	1111	15	1111

Table 1. Data Re-Ordering Table

2.5 Data Splitting

To use an Autoencoder network for anomaly detection, the data must be split into training and test data. There are two common approaches which are as follows,

- *Labeled Data:* The entire dataset can be characterized with both normal and abnormal data. It can then be split into training and test data using a traditional ratio, such as 60% training data and 40% test data. Some researchers use a 70/30 split. The ratio of the split is based on the researcher's judgement.
- *Unlabeled Data:* Only the normal data can be used to detect anomalies. This approach is used when the data is unlabeled. The faulty data is used as the test data.

In this research, the training dataset was obtained by plotting all the data to get the percentage of both normal and faulty signals.

2.6 Model Description

2.6.1 Autoencoder

Autoencoders are neural networks that learn to replicate its inputs at the output. It can be used for a variety of tasks, including dimensionality reduction, anomaly detection, and feature extraction.

In anomaly detection, autoencoders are used to identify data points that are significantly different from the rest of the data. This is done by comparing the output of the autoencoder to the input data. If the output is significantly different from the input, then the data point is likely to be an anomaly.

The anomaly score is calculated by measuring the difference between the output of an autoencoder and the input data. Data points with high anomaly scores are more likely to be anomalies.

Data points with anomaly scores that are higher than the

```
Dataset shape: (982, 4)
```

Figure 8. Shape of the Dataset Originally Collected in Two Dimension

threshold are flagged as anomalies.

2.6.2 Long Short-Term Memory Network (LSTM)

LSTM is a type of Recurrent Neural Network (RNN) that helps to retain long-term dependencies between data points in a sequence (Nguyen et al., 2020). This is achieved through the use of three control gates, such as the input gate, the output gate, and the forget gate. These gates are implemented using Sigmoid Neural Network layers and pointwise multiplication. LSTMs are typically implemented as a chain of repeated modules, each of which contains the three control gates.

LSTMs can read data sequentially as vector $S=\{S_1,S_2,S_3,\ldots..s_t,\ldots.\}$, where $S\in k^x$ represents the vector reading of x with x dimensions at time t. This Neural Network consists of three gates that helps to store relevant information and forget irrelevant ones. These gates are the forget gate, input gate and output gate.

2.6.3 LSTM-Autoencoder

An autoencoder is a neural network that can be used to detect anomalies by compressing large amounts of data into a smaller vector space. However, using an Autoencoder alone can be challenging for time series data, as the data can become very large over time, and the computation required to train and use the model can be prohibitive.

To address this challenge, LSTM-Autoencoders can be used. LSTMs are a type of recurrent neural network that are well-suited for processing time series data. Autoencoders can learn to remember long-term dependencies in data, which allows it to identify the most important features to remember. This reduces the amount of data that needs to be processed, which can significantly improve the performance of the model.

The architecture of an LSTM-Autoencoder is shown in Figure 9. The input data, which is a time series data x_u, x_d is first fed into the autoencoder. The autoencoder compresses the data into a smaller vector z. This compressed data is then fed into the LSTM. The LSTM learns to remember the important features of the data and forget the irrelevant features. This reduces the amount of data that needs to be reconstructed, which further improves the performance of the model.

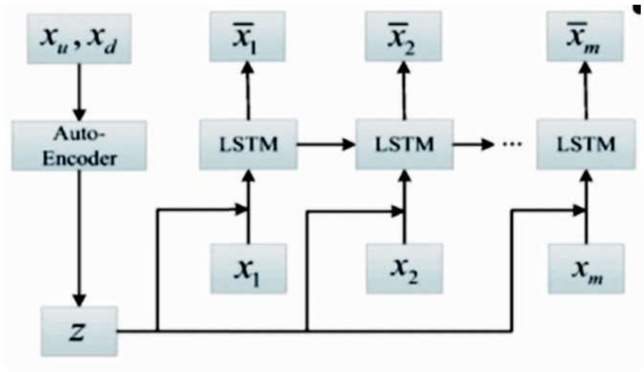

Figure 9. Architecture of LSTM-Autoencoder

During reconstruction, a threshold is selected. If the error between the reconstructed data and the original data is above the threshold, an anomaly is detected.

$$L = \frac{1}{2}\sum_{s} \|s - s^{\|} \times \|s - s^{\|} \qquad (2)$$

Equation 2 shows the LSTM-Autoencoder model. The pseudocode for the LSTM-Autoencoder is as follows,

LSTM-Autoencoder Algorithm

//INPUT: the training set Xu, //Xu is the dataset

//OUTPUT: prediction result X.

//Preprocess the data Xu to give Xuu

//Split Xuu to AutoEncoder training set XAE and XAD validation dataset.

//Initialize the weight matrices of AutoEncoder randomly.

//Put XAE into AutoEncoder.

if L(X, Y) < d then

Calculate the error L(X, Y)

Use the back propagation to train the AutoEncoder.

else

End the training.

end if

//Generate the characteristics of the input dataset Zt.

for t = 0 to epoch do

Put Zt into the LSTM, and do for forward propagation.

Generate output

Calculate error.

Use the back propagation to update parameters.

Use forward propagation to update network status ht,

end for

//Add LSTM after the encoder of AutoEncoder to form AE-LSTM.

//Fine-tuning the whole network, training initialization parameters.

//Input XAD test data in AE-LSTM to generate the predicted value X.

Return X.

2.6.4 Development of LSTM-Autoencoder Network with FFT

The model used in this research is the LSTM-Autoencoder with FFT. The LSTM-Autoencoder was used to aid the detection of anomalies in time series data. The FFT technique was also used to extract latent features of a signal in the frequency spectrum. The pseudocode for the model algorithm is as follows,

Model Algorithm (LSTM-Autoencoder with FFT)

//INPUT: the training set Xu, // Xu is the dataset

//OUTPUT: prediction result X.

//Preprocess the data Xu to give Xuu

//process the data using FFT to give Xuu'

//Split Xuu' to AutoEncoder training set XAE and XAD validation //dataset.

//Initialize the weight matrices of AutoEncoder randomly.

//Put XAE' into AutoEncoder.

if L(X, Y) < d then

Calculate the error L(X, Y)

Use the back propagation to train the AutoEncoder.

else

End the training.

end if

//Generate the characteristics of the input dataset Zt.

for t = 0 to epoch do

Put Zt into the LSTM, and do forward propagation.

Generate output

Calculate error.

Use the back propagation to update parameters.

Use forward propagation to update network status ht,

//INPUT: the training set Xu, // Xu is the dataset

//OUTPUT: prediction result X.

//Preprocess the data Xu to give Xuu

//Process the data using FFT to give Xuu'

//Split Xuu' to AutoEncoder training set XAE and XAD validation //dataset.

//Initialize the weight matrices of AutoEncoder randomly.

//Put XAE' into AutoEncoder.

if L(X, Y) < d then

Calculate the error L(X, Y)

Use the back propagation training the AutoEncoder.

else

End the training.

end if

//Generate the characteristics of the input dataset Zt.

for t = 0 to epoch do

Put Zt into the LSTM, and do for forward propagation.

Generate output

Calculate error.

Use the back propagation to update parameters.

Use forward propagation to update network status ht,

end for

//Add LSTM after the encoder of AutoEncoder to form AE-LSTM.

//Fine-tuning the whole network, training initialization parameters.

//Input XAD' test data in AE-LSTM to generate the predicted value X.

Return X.

2.6.5 Training and Testing

The model was trained with normal ball bearing data using 100 epochs and a batch size of 10. Afterwards, it was tested with data characterized by anomalies. The training loss and loss distribution were computed and plotted to serve as a guide for selecting a threshold. After the threshold was selected, the bearing failure plot was presented.

3. Results and Discussion

The data was pre-processed, converted from the time domain to the frequency domain, and features were extracted. Then, an LSTM-Autoencoder network was built and the loss was computed. The threshold was computed from the loss distribution, and the anomaly was observed graphically.

3.1 Data Description

The data used was obtained from the NASA repository. The data contains four files, each of which contains 20,800 data points read at a sampling rate of 20 kHz. Figure 10 shows the description of the dataset. The data set has 982 rows and 4 columns. The mean of the dataset is between 0.048 and 0.081. The standard deviation is within 0.009 to 0.040. The minimum value is 0.0007. The maximum value is 0.453.

3.2 Data Pre-Processing

The first step in the pre-processing stage is to merge the dataset. The dataset is currently stored in multiple files, each of which is labeled with a date and time stamp. Merging the files combines them into a single file, as shown in Figure 11. This allows the data to be used in the proposed model.

After the files have been merged, it is observed that the data was captured with date and time stamps. This means that the data is a sequence of data points, which can be used for LSTM-Autoencoders. LSTM-Autoencoders are a type of neural network that is specifically designed to learn from sequential data.

	Bearing 1	Bearing 2	Bearing 3	Bearing 4
count	982.000000	982.000000	982.000000	982.000000
mean	0.080951	0.078543	0.081351	0.047830
std	0.040200	0.011789	0.011607	0.009549
min	0.001168	0.000767	0.000716	0.001699
25%	0.060773	0.074240	0.076829	0.043951
50%	0.062021	0.075206	0.078187	0.044524
75%	0.083277	0.077458	0.080575	0.048130
max	0.453335	0.161016	0.151299	0.119047

Figure 10. Description of the Dataset

By merging the dataset and converting it into a sequence of data points, it can be easier for the LSTM-Autoencoder to learn from the data. This will improve the performance of the model and make it more accurate.

The data was divided into training data and test data. The training data is the data of the normal working bearing. The split was done so that the validation dataset would capture both normal and anomalous data, allowing anyone to observe the point at which the anomaly occurred. While the training data is mainly normal data, it is not possible to observe the normal data and the point of abnormality from the raw data. Additionally, since the data may be characterized by noise, Fast Fourier Transform (FFT) was employed to clean up the data and to observe the point suspected as anomalous.

3.3 Data Transformation using FFT

Figure 12(a) shows a graphical representation of the normal data (train data set) in the time domain. The time domain is a representation of the data as a function of time. In this case, the data is a series of numbers that represent the values of the signal at different points in time. Figure 12(a) shows that the data is relatively random, with no clear patterns. This makes it difficult to identify any anomalies in the data.

Figure 12(b) shows a graphical representation of the train data set in the frequency domain. The frequency domain is a representation of the data as a function of frequency. In this case, the data is a series of numbers that represent the amplitudes of the different frequencies that are present in the signal. Figure 12(b) shows that the data is

	Bearing 1	Bearing 2	Bearing 3	Bearing 4
2004-02-12 10:52:39	0.060236	0.074227	0.083926	0.044443
2004-02-12 11:02:39	0.061455	0.073844	0.084457	0.045081
2004-02-12 11:12:39	0.061361	0.075609	0.082837	0.045118
2004-02-12 11:22:39	0.061665	0.073279	0.084879	0.044172
2004-02-12 11:32:39	0.061944	0.074593	0.082626	0.044659

Figure 11. Data Representation of the Four Bearings Merged into One File

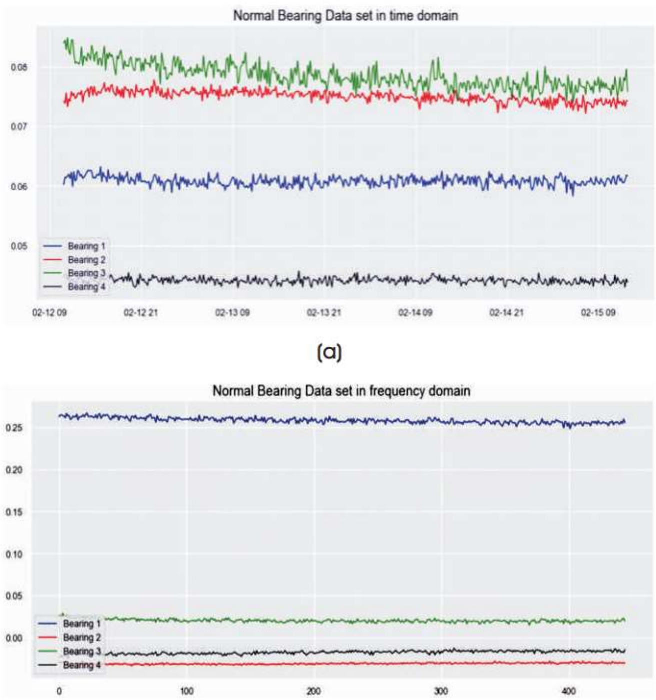

Figure 12. Graphical Representation of the Normal Data (a) in Time Domain (b) in Frequency Domain

less random in the frequency domain, with a few clear peaks. These peaks correspond to the frequencies that are most present in the signal. This makes it easier to identify any anomalies in the data, as they will stand out from the rest of the data.

Figures 13(a) and 13(b) show the graphical representation of the abnormal data in both time domain and frequency domain.

3.4 Further Pre-Processing

The LSTM-Autoencoder requires three-dimensional data as input. Therefore, the two-dimensional data shown in Figure 8 needs to be converted into three dimensions. This was done in time domain by repeating each data point three times, as shown in Figure 14.

3.5 LSTM-Autoencoder Network Used

After the data was reshaped as required by the LSTM-Autoencoder, the model was built and the data was fed into it. The summary of the LSTM-Autoencoder network is shown in Figure 15.

The LSTM network has 200 nodes in the first layer, 25 nodes in the second layer, and 200 nodes in the third layer. This configuration was chosen to achieve a lower threshold.

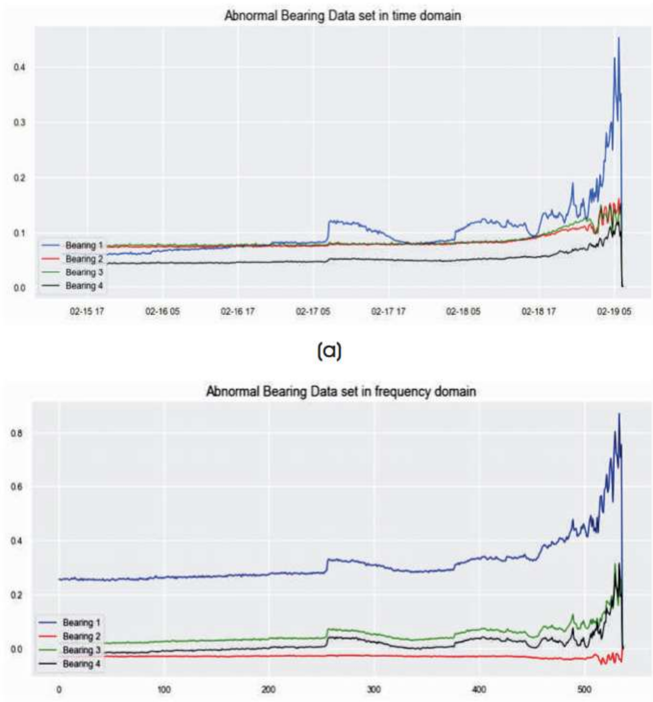

Figure 13. Graphical Representation of the Abnormal Data (a) in Time Domain (b) Frequency Domain

```
Training data shape: (445, 1, 4)
Test data shape: (538, 1, 4)
```

Figure 14. Shape of the Data Converted into Three-Dimension

Model: "model_3"

Layer (type)	Output Shape	Param #
input_3 (InputLayer)	(None, 1, 4)	0
lstm_11 (LSTM)	(None, 1, 200)	164000
lstm_12 (LSTM)	(None, 25)	22600
repeat_vector_3 (RepeatVecto	(None, 1, 25)	0
lstm_13 (LSTM)	(None, 1, 25)	5100
lstm_14 (LSTM)	(None, 1, 200)	180800
time_distributed_3 (TimeDist	(None, 1, 4)	804

Total params: 373,304
Trainable params: 373,304
Non-trainable params: 0

Figure 15. The Summary of the LSTM-Autoencoder Model

3.6 Loss Model

To evaluate the performance of the model, the loss is calculated. This was done when the network was trained for 100 epochs. Figure 16 represents the loss model and shows how much the training data differs from the validation data.

3.7 Loss Distribution

This parameter is needed to calculate the threshold. Figure 17 shows that the threshold can be below 0.126.

3.8 Anomaly Detection

Figure 18 shows that the model was able to detect anomalies at the selected threshold. The red line represents the threshold, and any signal above the red line is considered an anomaly.

3.9 Comparison of Results

In this paper, a new method for anomaly detection using FFT and LSTM-Autoencoder was proposed. The proposed method was compared to existing methods, such as Virtual Spectrum Imaging (VSI), Artificial Neural Network (ANN), and Stacked Denoising Autoencoder. The results showed that the proposed method was able to detect weak signals more accurately than the existing methods. Additionally, the proposed method was able to achieve a higher accuracy than the existing methods.

The researchers analyzed each dataset (RM2, RM3, RM4, and RM5) separately and observed different anomaly scores. For example, the anomaly score for RM2 was significantly higher than the anomaly scores for RM3, RM4, and RM5. This suggests that the proposed method is more sensitive to anomalies in some datasets than the others. Figures 19 and 20 show the anomaly and accuracy scores of the proposed system.

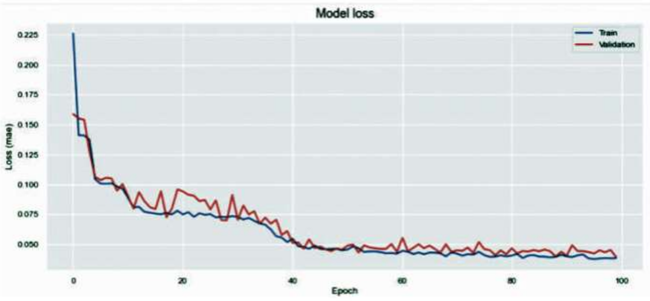

Figure 16. Loss Model

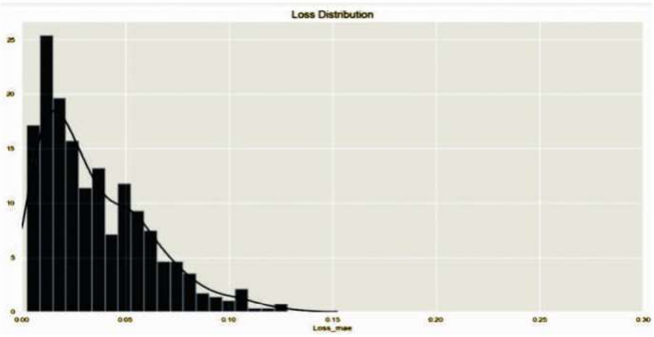

Figure 17. Loss Distribution

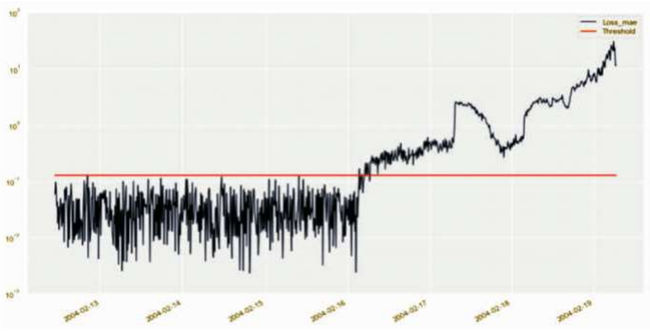

Figure 18. Anomaly Detection

Conclusion and Recommendations

This research focuses on detecting anomalies in motor bearings. Although several research studies have been conducted to find techniques that would better assist in anomaly detection, the goal of this research is to detect anomalies in weak signals. To achieve this, the Long Short-Term Memory (LSTM)-Autoencoder (AE) with Fast Fourier Transform (FFT) was used. FFT was used to clean up and transform the data to reduce complex multiplications, speed up computation, and identify if an anomaly exists,

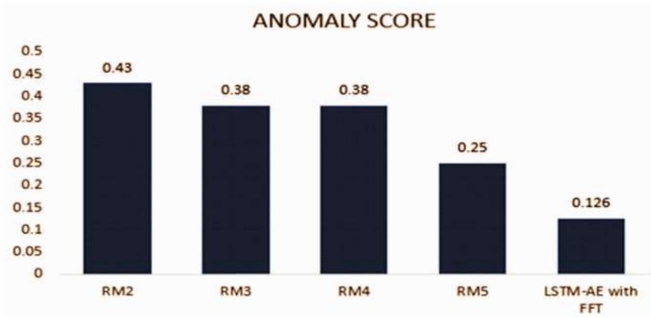

Figure 19. Anomaly Score of Bearing Data

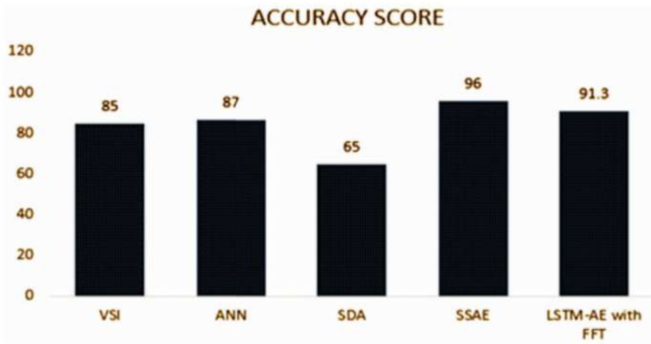

Figure 20. Accuracy Score

while the LSTM-Autoencoder was used to validate and detect the anomaly before it occurs. As a result of using Fast Fourier Transform (FFT), the LSTM-Autoencoder technique was able to detect anomalies at a low threshold of 0.126, compared to all other techniques that have been used. The research has been able to improve the sensitivity of detecting anomalies in motor bearings. Better sensitivity was achieved with the use of FFT and LSTM-Autoencoder. Sparse Stack Autoencoder with FFT can improve early detection of anomalies by lowering the threshold and improving accuracy.

Future Work

In the future, in addition to detection anomaly localization can be achieved through the use of Machine Learning and statistical models. This will help to expedite the maintenance process.

References

[1]. Abid, F. B., Sallem, M., & Braham, A. (2019, March). Optimized SWPT and decision tree for incipient bearing fault diagnosis. In 2019 *19th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)* (pp. 231-236). IEEE. https://doi.org/10.1109/STA.2019.8717197

[2]. Abouelanouar, B., Elamrani, M., Elkihel, B., & Delaunois, F. (2018). Application of wavelet analysis and its interpretation in rotating machines monitoring and fault diagnosis - A review. *International Journal of Engineering & Technology,* 7(4), 3465-3471.

[3]. Adewumi, A. O., & Akinyelu, A. A. (2017). A survey of machine-learning and nature-Inspired based credit card fraud detection techniques. *International Journal of System Assurance Engineering and Management,* 8, 937-953. https://doi.org/10.1007/s13198-016-0551-y

[4]. Ahmad, S., Styp-Rekowski, K., Nedelkoski, S., & Kao, O. (2020, December). Autoencoder-based condition monitoring and anomaly detection method for rotating machines. In 2020 *IEEE International Conference on Big Data (Big Data)* (pp. 4093-4102). IEEE. https://doi.org/10.1109/BigData50022.2020.9378015

[5]. Asakura, T., Yashima, W., Suzuki, K., & Shimotou, M. (2020). Anomaly detection in a logistic operating system using the Mahalanobis–Taguchi method. *Applied Sciences,* 10(12), 4376. https://doi.org/10.3390/app10124376

[6]. Barot, A., & Kulkarni, P. (2021). Technological evolution in the fault diagnosis of rotating machinery: A review. *IOSR Journal of Mechanical and Civil Engineering,* 18(2), 9-26.

[7]. Boniol, P., Linardi, M., Roncallo, F., & Palpanas, T. (2020, April). Automated anomaly detection in large sequences. In 2020 *IEEE 36th International Conference on Data Engineering (ICDE)* (pp. 1834-1837). IEEE. https://doi.org/10.1109/ICDE48307.2020.00182

[8]. Chalapathy, R., & Chawla, S. (2019). Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407.* https://doi.org/10.48550/arXiv.1901.03407

[9]. Cooper, C., Zhang, J., Gao, R. X., Wang, P., & Ragai, I. (2020). Anomaly detection in milling tools using acoustic signals and generative adversarial networks. *Procedia Manufacturing,* 48, 372-378. https://doi.org/10.1016/j.promfg.2020.05.059

[10]. Egaji, O. A., Ekwevugbe, T., & Griffiths, M. (2020, July). A data mining based approach for electric motor anomaly detection applied on vibration data. In 2020 *Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)* (pp. 330-334). IEEE. https://doi.org/10.1109/WorldS450073.2020.9210318

[11]. Glowacz, A. (2019). Fault diagnosis of single-phase induction motor based on acoustic signals. *Mechanical Systems and Signal Processing,* 117, 65-80. https://doi.org/10.1016/j.ymssp.2018.07.044

[12]. Gong, C. S. A., Lee, H. C., Chuang, Y. C., Li, T. H., Su, C. H. S., Huang, L. H., ... & Chang, C. H. (2018). Design and implementation of acoustic sensing system for online early fault detection in industrial fans. *Journal of Sensors,* 1-15. https://doi.org/10.1155/2018/4105208

[13]. Hruntovich, N. V., Kapanski, A. A., Petrov, I. V., & Kostyleva, E. E. (2019). Vibration diagnostic of electric motor roller bearings. In *E3S Web of Conferences,* 124, 02008. EDP Sciences. https://doi.org/10.1051/e3sconf/201912402008

[14]. Jabczyński, J., & Szczesniak, A. (1995). Digital processing of Doppler signals using fast Fourier transform. *Optica Applicata,* 25(4), 281-289.

[15]. Jiang, W., Hong, Y., Zhou, B., He, X., & Cheng, C. (2019). A GAN-based anomaly detection approach for imbalanced industrial time series. *IEEE Access,* 7, 143608-143619. https://doi.org/10.1109/ACCESS.2019.2944689

[16]. Jin, X., Que, Z., Sun, Y., Guo, Y., & Qiao, W. (2019). A data-driven approach for bearing fault prognostics. *IEEE Transactions on Industry Applications,* 55(4), 3394-3401. https://doi.org/10.1109/TIA.2019.2907666

[17]. Kanwal, N., Girdhar, A., Kaur, L., & Bhullar, J. S. (2019, April). Detection of digital image forgery using fast Fourier transform and local features. In 2019 *International Conference on Automation, Computational and Technology Management (ICACTM)* (pp. 262-267). IEEE. https://doi.org/10.1109/ICACTM.2019.8776709

[18]. Khadersab, A., & Shivakumar, S. (2018). Vibration analysis techniques for rotating machinery and its effect on bearing faults. *Procedia Manufacturing,* 20, 247-252. https://doi.org/10.1016/j.promfg.2018.02.036

[19]. Kwon, D., Kim, H., Kim, J., Suh, S. C., Kim, I., & Kim, K. J. (2019). A survey of deep learning-based network anomaly detection. *Cluster Computing,* 22, 949-961. https://doi.org/10.1007/s10586-017-1117-8

[20]. Lee, G. Y., Kim, M., Quan, Y. J., Kim, M. S., Kim, T. J. Y., Yoon, H. S., ... & Ahn, S. H. (2018a). Machine health management in smart factory: A review. *Journal of Mechanical Science and Technology,* 32, 987-1009. https://doi.org/10.1007/s12206-018-0201-1

[21]. Lee, H., Li, G., Rai, A., & Chattopadhyay, A. (2020). Real-time anomaly detection framework using a support vector regression for the safety monitoring of commercial aircraft. *Advanced Engineering Informatics,* 44, 101071. https://doi.org/10.1016/j.aei.2020.101071

[22]. Lee, K., Kim, J. K., Kim, J., Hur, K., & Kim, H. (2018b, July). CNN and GRU combination scheme for bearing anomaly detection in rotating machinery health monitoring. In 2018 *1st IEEE International Conference on Knowledge Innovation and Invention (ICKII)* (pp. 102-105). IEEE. https://doi.org/10.1109/ICKII.2018.8569155

[23]. Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis,* 42, 60-88. https://doi.org/10.1016/j.media.2017.07.005

[24]. Liu, C., & Gryllias, K. (2020). A semi-supervised Support Vector Data Description-based fault detection method for rolling element bearings based on cyclic spectral analysis. *Mechanical Systems and Signal Processing,* 140, 106682. https://doi.org/10.1016/j.ymssp.2020.106682

[25]. Mohammadi, M., Al-Fuqaha, A., Sorour, S., & Guizani, M. (2018). Deep learning for IoT big data and streaming analytics: A survey. *IEEE Communications Surveys & Tutorials,* 20(4), 2923-2960. https://doi.org/10.1109/COMST.2018.2844341

[26]. Nath, S. (2020). *Low Latency Anomaly Detection with Imperfect Models* (Doctoral dissertation, University of Arkansas).

[27]. Pandarakone, S. E., Masuko, M., Mizuno, Y., & Nakamura, H. (2018, September). Deep neural network based bearing fault diagnosis of induction motor using fast Fourier transform analysis. In 2018 *IEEE Energy Conversion Congress and Exposition (ECCE)* (pp. 3214-3221). IEEE. https://doi.org/10.1109/ECCE.2018.8557651

[28]. Pittino, F., Puggl, M., Moldaschl, T., & Hirschl, C. (2020). Automatic anomaly detection on in-production manufacturing machines using statistical learning methods. *Sensors,* 20(8), 2344. https://doi.org/10.3390/s20082344

[29]. Saeki, M., Ogata, J., Murakawa, M., & Ogawa, T. (2019, June). Visual explanation of neural network based rotation machinery anomaly detection system. In 2019 *IEEE International Conference on Prognostics and Health Management (ICPHM)* (pp. 1-4). IEEE. https://doi.org/10.1109/ICPHM.2019.8819396

[30]. Schmidt, S., & Heyns, P. S. (2019). Localised gear anomaly detection without historical data for reference density estimation. *Mechanical Systems and Signal Processing,* 121, 615-635. https://doi.org/10.1016/j.ymssp.2018.11.051

[31]. Sohaib, M., & Kim, J. M. (2018). Reliable fault diagnosis of rotary machine bearings using a stacked sparse autoencoder-based deep neural network. *Shock and Vibration,* 2018, 1-11. https://doi.org/10.1155/2018/2919637

[32]. Sohaib, M., & Kim, J. M. (2019). Fault diagnosis of rotary machine bearings under inconsistent working conditions. *IEEE Transactions on Instrumentation and Measurement,* 69(6), 3334-3347. https://doi.org/10.1109/TIM.2019.2933342

[33]. Sulka, P., Sapietova, A., Dekys, V., & Stalmach, O. (2019). Vibration analysis and comparison of the damaged and undamaged rolling ball bearing. *Transportation Research Procedia,* 40, 511-518. https://doi.org/10.1016/j.trpro.2019.07.074

[34]. Vo, Q., Fang, F., Zhang, X., & Gao, H. (2017). Surface recovery algorithm in white light interferometry based on combined white light phase shifting and fast Fourier transform algorithms. *Applied Optics,* 56(29), 8174-8185. https://doi.org/10.1364/AO.56.008174

[35]. Wang, Z., Zhou, Y., & Li, G. (2019, November). Anomaly detection for machinery by using Big Data Real-Time processing and clustering technique. In *Proceedings of the 3rd International Conference on Big Data Research* (pp. 30-36). https://doi.org/10.1145/3372454.3372480

[36]. Wei, W., Wu, H., & Ma, H. (2019). An autoencoder and LSTM-based traffic flow prediction method. *Sensors,* 19(13), 2946. https://doi.org/10.3390/s19132946

[37]. Zhang, S., Ye, F., Wang, B., & Habetler, T. G. (2019). Semi-supervised learning of bearing anomaly detection via deep variational autoencoders. *arXiv preprint arXiv:1912.01096.* https://doi.org/10.48550/arXiv.1912.01096

ABOUT THE AUTHORS

Sulaimon A. Bashir, Department of Computer Science, School of Information and Communication Technology, Federal University of Technology, Minna, Nigeria.

Oladebo Suliat Jimoh, Department of Computer Science, School of Information and Communication Technology, Federal University of Technology, Minna, Nigeria.

Idris Mohammed Kolo, Department of Computer Science, School of Information and Communication Technology, Federal University of Technology, Minna, Nigeria.

Enesi Femi Aminu, Department of Computer Science, School of Information and Communication Technology, Federal University of Technology, Minna, Nigeria.

i-manager Publications

3/343, Hill view, Town Railway Nager, Nagercoil
Kanyakumari Dist. Pin-629 001.
Tel: +91-4652-231675, 232675, 276675

e-mail: info@imanagerpublications.com
contact@imanagerpublications.com
www.imanagerpublications.com