



DOCTORAL DISSERTATION

in the discipline: Control, Electronics and Electrical Engineering

Vehicle Diagnostics using Artificial Intelligence and Digital Signal Processing Methods

Author: Muhammad AHSAN

Supervisor: Prof. Pol. Śl. Dariusz BISMOR, PhD, DSc, Eng.

Department of Measurements and Control Systems
Faculty of Automatic Control, Electronics, and Computer Science
Silesian University of Technology

Gliwice, February 2025

DOCTORAL DISSERTATION

in the discipline: Control, Electronics and Electrical Engineering

Vehicle Diagnostics Using Artificial Intelligence and Digital Signal Processing Methods

Author:

Muhammad AHSAN

Supervisor:

Prof. Pol. Śl. Dariusz BISMOR, PhD, DSc, Eng.

Silesian University of Technology

Gliwice, February 2025

Author Declaration

I declare that the doctoral dissertation entitled **Vehicle Diagnostics using Artificial Intelligence and Digital Signal Processing Methods** is consistent with the doctoral dissertation prepared and submitted as part of education at the doctoral school run by the Silesian University of Technology.

Muhammad Ahsan,

Department of Measurements and Control Systems,
Faculty of Automatic Control, Electronics and Computer Science,
Silesian University of Technology, Gliwice, Poland
February 09, 2025

Acknowledgments

I am deeply grateful for the opportunity to have been surrounded by many inspiring individuals throughout my time in Gliwice, Poland. This section provides me with the privilege of expressing my sincere appreciation to all those whose invaluable support, guidance, and encouragement have contributed to the successful completion of this dissertation.

First and foremost, I extend my deepest gratitude to my supervisor, **Prof. Dr. Hab. Eng. Dariusz Bismor**, for his unwavering support, insightful guidance, and encouragement, both academically and personally. His extensive expertise, profound knowledge, and passion for research have been invaluable to my academic journey. His mentorship has greatly shaped my research, from its initial phases to the completion of this dissertation. I am especially grateful for the professional and nurturing research environment he has fostered, where every student is treated with care and respect. I consider myself truly fortunate to have had the privilege of his supervision.

I would also like to express my sincere appreciation to **Prof. Dr. Hab. Eng. Marek Pawełczyk**, Head of the Department of Measurements and Control Systems and Rector of the Silesian University of Technology, for his leadership and dedication to fostering an outstanding academic and research environment. His commitment to excellence in education and research has greatly contributed to my academic development, and I am truly grateful for his support and the opportunities provided by the university under his guidance.

I extend my sincere gratitude to **Dr. Eng. Dariusz Buchczik** for his guidance and technical support in hardware and calibration at the Department of Measurements and Control Systems, Silesian University of Technology. I am also deeply thankful to **Dr. Hab. Eng. Paweł Fabiś** for his assistance in vehicle data acquisition at the Department of Road Transport, Silesian University of Technology. Their support has been invaluable in the experimental aspects of my research.

I am also grateful to **Eng. M. Arslan** from MBZUAI, Abu Dhabi, and **Prof. Marco Cocconcelli** from UNIMORE, Italy, for their helpful insights on machine learning and data preprocessing. Their assistance was appreciated in certain aspects of my research.

Additionally, I extend my heartfelt appreciation to **Dr. Eng. Oliwia Krauze**

and **Eng. Faisal Saleem**, my esteemed colleagues, for their invaluable support at the Department of Measurements and Control Systems, Silesian University of Technology, Poland. Their collaboration, encouragement, and unwavering support have been instrumental throughout my academic journey, both professionally and personally.

On a more personal note, I would like to express my gratitude to the many incredible people and fellow students—far too numerous to name individually—whom I have had the pleasure of meeting over the years in Gliwice. Their friendship and camaraderie have made it easy to maintain a healthy balance between work and personal life.

Furthermore, it has been an honor to spend the past four years at the Silesian University of Technology, an institution that, in my experience, offers one of the best research environments I have encountered. The university's high-profile yet welcoming and open-minded atmosphere, along with its supportive academic culture, has been invaluable to my development. In particular, I extend my gratitude to all the teachers at the **Doctoral School**, whose encouragement and support meant a great deal to me.

Finally, I am forever indebted to my parents and siblings for their unconditional love and unwavering support. Words cannot adequately express the gratitude I feel toward them. I would like to conclude this acknowledgment with a quote: "*Happiness is only real when shared.*" I truly believe in these words, and as I reflect on the past four years, I am profoundly grateful to all those with whom I have shared this journey.

Contents

List of Figures	XI
List of Tables	XII
Abstract (English)	XIV
Abstract (Polish)	XVI
List of Publications	XVIII
1 Introduction	1
1.1 Background and Motivation	1
1.2 Literature Review	4
1.2.1 MEMS Accelerometers and Embedded Platforms	4
1.2.2 Signal Processing Methods for Fault Diagnosis	6
1.2.3 Machine Learning Methods for Fault Diagnosis	19
1.3 Research Objectives	21
1.4 Contributions	22
1.5 Structure of the Dissertation	23
2 Interfacing and Calibration of the ADXL1002 Accelerometer with BeagleBone Black	24
2.1 Introduction	24
2.2 Interfacing	25
2.2.1 ADXL1002 Accelerometer	25
2.2.2 BeagleBone Black	27
2.2.3 Interfacing Results	28
2.3 Calibration	31
2.3.1 Laboratory Setup	31
2.3.2 Vibration Dataset	32
2.3.3 Calibration Results	35
2.4 Conclusion	38

3	Vibration Datasets	40
3.1	Introduction	40
3.2	Vibration Dataset from Vehicle Engine using ADXL1002 Accelerometer	41
3.2.1	Experimental Setup	41
3.2.2	Vehicle Engine Dataset	43
3.3	Bearing Vibration Datasets in the Literature	45
3.3.1	IMF Test-to-Failure Vibration Dataset	45
3.3.2	MFPT Bearing Fault Dataset	46
3.4	Conclusion	48
4	Digital Signal Processing Methods for Vehicle Diagnostics	49
4.1	Introduction	49
4.2	Overview of Digital Signal Processing Methods	50
4.2.1	DSP Methods for Bearing Faults	51
4.2.2	DSP Methods for Engine Misfire	51
4.2.3	Challenges in DSP Methods for Vehicle Diagnostics	52
4.3	HSA-based BPF Optimization	52
4.3.1	Implementation of HSA	53
4.3.2	Simulation Results for HSA-based BPF	56
4.4	PSO-based ARL-Wavelet BPF Optimization	63
4.4.1	Implementation of PSO Algorithm:	65
4.4.2	Simulation Results for PSO-based ARL-Wavelet BPF	67
4.5	IMF-based Vehicle Engine Misfire Diagnostics	69
4.5.1	Implementation of IMF	70
4.5.2	Simulation Results for IMF-based Misfire Diagnostics	72
4.6	Comparative Analysis of DSP Methods	77
4.6.1	Diagnostic Performance of Filter Design Approaches	77
4.6.2	Diagnostic Performance of FFT, ES, and EMD Approaches	77
4.7	Conclusion	78
5	Artificial Intelligence Methods for Vehicle Diagnostics	80
5.1	Introduction	80
5.2	Overview of Artificial Intelligence Methods	82
5.3	Dataset and Preprocessing	83
5.3.1	Vibration Dataset	83
5.3.2	Preprocessing	84
5.4	DCNN Models for Vehicle Diagnostics	86
5.4.1	Implementation of DCNN Models	87
5.4.2	Simulation Results for 1D and 2D Datasets	90
5.5	DCNN-LSTM Models for Vehicle Diagnostics	95

5.5.1	Implementation of DCNN-LSTM Models	96
5.5.2	Simulation Results for 1D and 2D Datasets	99
5.6	multi-Kernel-DCNN-LSTM Models for Vehicle Diagnostic	104
5.6.1	Implementation of multi-Kernel-DCNN-LSTM Models	104
5.6.2	Simulation Results for 1D and 2D Datasets	105
5.7	Comparative Analysis of AI Methods	109
5.7.1	Performance Analysis	110
5.7.2	Impact of 1D and 2D Data	110
5.7.3	Challenges and Limitations	111
5.8	Conclusion	111
6	Conclusions	113
6.1	Interfacing and Calibration of the ADXL1002 Accelerometer	114
6.2	Vibration Data Acquisition and Preprocessing	114
6.3	Development of DSP Methods	115
6.4	Development of AI Methods	115
6.5	Comparative Analysis	116
6.6	Challenges and Advantages of the ADXL1002 Accelerometer	117
6.7	Contributions and Implications of Research	117
6.8	Limitations and Future Work	118
A	ADXL1002: Performance, Bandwidth, and Features	121
A.1	Optimal Performance of the ADXL1002	121
A.2	Operation of the ADXL1002	122
A.3	Output Signal Bandwidth of the ADXL1002	122
A.4	Features of the ADXL1002	123
B	BeagleBone Black: Specifications and Component	125
B.1	I/Os of the BeagleBone Black	125
B.2	LEDs of the BeagleBone Black	125
B.3	Networking of the BeagleBone Black	125
B.4	Power Management of the BeagleBone Black	126
B.5	Features of the BeagleBone Black	126
C	Bearing Fault Frequencies	128
C.1	Parameters	128
C.2	Fault Frequencies	128
C.3	Steps to Compute	129

D	Converting 1D Data to 2D Grayscale Images	130
D.1	Initialization	130
D.2	Iterating Through Fault Categories	130
D.3	Segmenting and Reshaping	131
D.4	Reshaping 1D to 2D	131
D.5	Label Encoding and One-Hot Encoding	131
D.6	Summary of Workflow	132
E	t-SNE Implementation	133
E.1	Load dataset and ensure correct data type	133
E.2	Apply t-SNE for dimensionality reduction	133
E.3	Create a DataFrame for visualization	134
E.4	Plot the 2D t-SNE components	134
F	Accuracy and Loss of 10 Runs	135
F.1	DCNN model for 1D and 2D dataset	135
F.2	DCNN-LSTM model for 1D and 2D dataset	137
F.3	Multi-Kernel DCNN-LSTM model for 1D and 2D dataset	139

References

List of Figures

1.1	A graph showing global vehicle production per 1,000 people with a highlight on Poland's position in Europe.	1
1.2	Vibration signal in the time domain: $x(t) = A \sin(2\pi ft + \phi)$ with $A = 1.0$, $f = 5.0$ Hz, and $\phi = \pi/4$	7
1.3	Complex vibration signal in the time domain: $x(t)$ with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N	7
1.4	RMS of the complex vibration signal: $x(t)$ with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N	8
1.5	FFT of the complex vibration signal: $x(t)$ with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N	11
1.6	Complex vibration signal and its envelope with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N	12
1.7	ES of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N . . .	13
1.8	EMD of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N . . .	15
1.9	STFT of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N . . .	17
1.10	STFT-based-SK of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N	18
1.11	PSD of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N . . .	19
2.1	EVAL-ADXL1002Z accelerometer and its function block diagram.	26
2.2	BeagleBone Black with its key components.	27

2.3	Block diagram representing interface of ADXL1002 accelerometer with BeagleBone Black and its power module.	28
2.4	Power supply module.	29
2.5	Filter with voltage divider.	30
2.6	LED and switch module.	30
2.7	Laboratory set for calibration of ADXL1002 accelerometer.	32
2.8	Dataset recorded with ADXL1002 accelerometer at $1 m/s^2$ acceleration.	33
2.9	Dataset recorded with ADXL1002 accelerometer at various acceleration.	33
2.10	Time-domain representation of the dataset recorded with ADXL1002 accelerometer.	34
2.11	Frequency-domain representation of the dataset recorded with ADXL1002 accelerometer.	35
2.12	Relationship graph between reference frequencies of the vibration shaker and measured frequencies of the ADXL1002 accelerometer.	36
2.13	Errors among reference frequencies of the vibration shaker and measured frequencies of the ADXL1002 accelerometer.	37
2.14	Sensitivities of both accelerometers at different frequencies a) sensitivity of the first accelerometer and b) sensitivity of the second accelerometer.	37
2.15	Sensitivity versus acceleration at different frequencies: a) characteristic of the first accelerometer, and b) characteristic of the second accelerometer.	38
3.1	Placement of ADXL1002 Accelerometer on Vehicle Engine for Vibration Data Collection.	41
3.2	Placement of ADXL1002 Accelerometer on Vehicle Engine for Vibration Data Collection.	42
3.3	3D representation of the vehicle vibration dataset recorded with ADXL1002 accelerometer.	44
3.4	Time-domain representation of the vehicle vibration dataset recorded with ADXL1002 accelerometer.	44
3.5	Experimental setup for IMF Test-to-Failure Vibration Dataset Recording [1].	46
3.6	IMF Test-to-Failure Dataset Visualization.	47
4.1	Flow diagram of proposed HSA.	54
4.2	Raw vibration signal 1515 in (a) time-domain and (b) its ES from IMS bearing dataset with FTF fault.	57
4.3	For FTF fault signal 1515; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.	57

4.4	Raw vibration signal 1598 in (a) time-domain and (b) its ES from IMS bearing dataset with FTF fault.	58
4.5	For FTF fault signal 1598; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.	58
4.6	Raw vibration signal 6075 in (a) time-domain and (b) its ES from IMS bearing dataset with OR fault.	59
4.7	For OR fault signal 6075; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.	59
4.8	Raw vibration signal 6100 in (a) time-domain and (b) its ES from IMS bearing dataset with OR fault.	60
4.9	For OR fault signal 6100; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.	60
4.10	Raw vibration signal 411 in (a) time-domain and (b) its ES from MFPT dataset with OR fault.	61
4.11	For OR fault signal 411; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.	61
4.12	Raw vibration signal 413 in (a) time-domain and (b) its ES from MFPT dataset with OR fault.	62
4.13	For OR fault signal 413; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.	62
4.14	ARL-Wavelet with $f_c = 20Hz$ and $\sigma = 5Hz$ and $10Hz$ a) Time and b) frequency plots.	64
4.15	First OR fault from MFPT dataset: a) time-domain vibration signal, b) filtered time-domain signal, c) ES of unfiltered signal, and d) ES of filtered signal.	68
4.16	Second OR fault from MFPT dataset: a) time-domain vibration signal, b) filtered time-domain signal, c) ES of unfiltered signal, and d) ES of filtered signal.	68
4.17	First baseline fault from MFPT dataset: a) time-domain vibration signal, b) filtered time-domain signal, c) ES of unfiltered signal, and d) ES of filtered signal.	68
4.18	Second baseline fault from MFPT dataset: a) time-domain vibration signal, b) filtered time-domain signal, c) ES of unfiltered signal, and d) ES of filtered signal.	69

- 4.19 FFT of the vibration signals recorded from the vehicle engine using ADXL1002 accelerometer interfaced with BBB: a & b) FFT of the healthy signals without load and misfire conditions, c, d, e, f, g, & h) FFT of the healthy signals with load and without misfire conditions, and i, j, k, l, m, & n) FFT of the faulty signals with load and with misfire conditions. 73
- 4.20 ES of the healthy vibration signals recorded from the vehicle engine using ADXL1002 accelerometer interfaced with BBB: a) ES of the healthy signal recorded at 1500 RPMs (25 Hz) with no load, b) ES of the healthy signal recorded at 3000 RPMs (50 Hz) with no load, c & d) ES of the healthy signals recorded at 1500 RPMs (25 Hz) with misfire conditions at half load and full load, e & f) ES of the healthy signals recorded at 2500 RPMs (41.67 Hz) with misfire conditions at half load and full load, and g & h) ES of the healthy signals recorded at 3000 RPMs (50 Hz) with misfire conditions at half load and full load. 74
- 4.21 ES of the misfired vibration signals recorded from the vehicle engine using ADXL1002 accelerometer interfaced with BBB: a & b) ES of the faulty signals recorded at 1500 RPMs (25 Hz) with misfire conditions at half load and full load, c & d) ES of the faulty signals recorded at 2500 RPMs (41.67 Hz) with misfire conditions at half load and full load, and e & f) ES of the faulty signals recorded at 3000 RPMs (50 Hz) with misfire conditions at half load and full load. 74
- 4.22 Comparison of healthy and misfire signals using FFT of the first-IMF: a) EMD of the healthy vibration signal with load at 2500 RPMs, b) EMD of the misfire vibration signal with load at 2500 RPMs, c) FFT of the first-IMF from (a), and d) FFT of the first-IMF from (b). 75
- 4.23 FFT of the first-IMF: a & b) healthy signals with 1500 (25 Hz) and 3000 RPMs (50 Hz) without misfire conditions and unload, c & d) healthy signals with 1500 RPMs (25 Hz) without misfire conditions at half load and full load, e & f) healthy signals with 2500 RPMs (41.67 Hz) without misfire conditions at half load and full load, and g & h) healthy signals with 3000 RPMs (50 Hz) without misfire conditions at half load and full load. 76
- 4.24 FFT of the first-IMF: a & b) faulty signals with 1500 RPMs (25 Hz) and with misfire conditions at half load and full load, c & d) faulty signals with 2500 RPMs (41.67 Hz) and with misfire conditions at half load and full load, and e & f) faulty signals with 3000 RPMs (50 Hz) and with misfire conditions at half load and full load. 76

5.1	Block diagram for preprocessing steps.	84
5.2	First-IMF associated with each health condition of the vehicle engine: a) healthy condition, b) misfire condition at 1500 RPM, c) misfire condition at 2500 RPM, and d) misfire condition at 3000 RPM.	85
5.3	Random 1D vibration signals after segmentation and shuffling with a window length of 4096 samples.	85
5.4	Random 2D grayscale images after segmentation and shuffling with a window length of 4096 samples.	85
5.5	t-SNE visualization of vibration data before training.	86
5.6	Proposed models for misfire diagnosis in vehicle engine.	87
5.7	Proposed 2D DCNN model for misfire diagnosis in vehicle engine.	87
5.8	Training and validation accuracies over 10 independent runs of 1D DCNN model.	91
5.9	Training and validation accuracies over 10 independent runs of 2D DCNN model.	91
5.10	Best training accuracy and validation accuracy for: a) 1D DCNN model, b) 2D DCNN model.	92
5.11	Confusion matrices representing true fault versus predicted fault for engine misfire at different RPMs: a) 1D DCNN model, and b) 2D DCNN model.	93
5.12	t-SNE visualization of vibration data after training; a) 1D DCNN model, and b) 2D DCNN model.	93
5.13	Performance results of 1D DCNN model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.	95
5.14	Performance results of 2D DCNN model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.	95
5.15	Proposed 2D DCNN-LSTM model for misfire diagnosis in vehicle engine.	96
5.16	Training and validation accuracies over 10 independent runs of 1D DCNN-LSTM model.	99
5.17	Training and validation accuracies over 10 independent runs of 2D DCNN-LSTM model.	100
5.18	Best training accuracy and validation accuracy for: a) 1D DCNN-LSTM model, b) 2D DCNN-LSTM model.	100
5.19	Confusion matrices representing true fault versus predicted fault for engine misfire at different RPMs: a) 1D DCNN-LSTM model, b) 2D DCNN-LSTM model.	101
5.20	t-SNE visualization of vibration data after training; a) 1D DCNN-LSTM model, b) 2D DCNN-LSTM model.	101

5.21	Performance results of 1D DCNN-LSTM model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.	103
5.22	Performance results of 2D CNN-LSTM model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.	103
5.23	Proposed multi-Kernel-2D-DCNN-LSTM model for misfire diagnosis in vehicle engine.	104
5.24	Training and validation accuracies over 10 independent runs for 1D dataset.	105
5.25	Training and validation accuracies over 10 independent runs for 2D dataset.	105
5.26	Best training accuracy and validation accuracy for: a) 1D multi-Kernel-DCNN-LSTM model, b) 2D multi-Kernel-DCNN-LSTM model.	106
5.27	Confusion matrices representing true fault versus predicted fault for engine misfire at different RPMs: a) 1D multi-Kernel-DCNN-LSTM model, b) 2D multi-Kernel-DCNN-LSTM model.	107
5.28	t-SNE visualization of vibration data after training: a) 1D multi-Kernel-DCNN-LSTM model, b) 2D multi-Kernel-DCNN-LSTM model.	107
5.29	Performance results of 1D multi-Kernel-DCNN-LSTM model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.	108
5.30	Performance results of 2D multi-Kernel-DCNN-LSTM model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.	108
5.31	Comparison of maximum accuracy, minimum accuracy, and average accuracy for the 10 independent runs for different models with 1D and 2D datasets.	109
A.1	Application circuit of ADXL1002 accelerometer.	121
A.2	Frequency response of ADXL1002 accelerometer [2].	123
F.1	Training accuracy and training loss of 10 runs using 1D DCNN model.	135
F.2	Validation accuracy and validation loss of 10 runs using 1D DCNN model.	136
F.3	Best model training and validation accuracy and loss for 1D DCNN model.	136
F.4	Training accuracy and training loss of 10 runs using 2D DCNN model.	136
F.5	Validation accuracy and validation loss of 10 runs using 2D DCNN model.	137
F.6	Best model training and validation accuracy and loss for 2D DCNN model.	137
F.7	Training accuracy and training loss of 10 runs using 1D DCNN-LSTM model.	137
F.8	Validation accuracy and validation loss of 10 runs using 1D DCNN-LSTM model.	138
F.9	Best model training and validation accuracy and loss for 1D DCNN-LSTM model.	138
F.10	Training accuracy and training loss of 10 runs using 2D DCNN-LSTM model.	138

F.11 Validation accuracy and validation loss of 10 runs using 2D DCNN-LSTM model.	139
F.12 Best model training and validation accuracy and loss for 2D DCNN-LSTM model.	139
F.13 Training accuracy and training loss of 10 runs using 1D multi-Kernel DCNN-LSTM model.	139
F.14 Validation accuracy and validation loss of 10 runs using 1D multi-Kernel DCNN-LSTM model.	140
F.15 Best model training and validation accuracy and loss for 1D multi-Kernel DCNN-LSTM model.	140
F.16 Training accuracy and training loss of 10 runs using 2D multi-Kernel DCNN-LSTM model.	140
F.17 Validation accuracy and validation loss of 10 runs using 2D multi-Kernel DCNN-LSTM model.	141
F.18 Best model training and validation accuracy and loss for 2D multi-Kernel DCNN-LSTM model.	141

List of Tables

2.1	ADXL1002 pin function description.	26
2.2	List of Hardware and Description.	31
3.1	Vehicle engine vibration dataset.	43
4.1	Hyperparameters of HSA.	55
4.2	Filter parameters for IMS FTF faults diagnosis.	58
4.3	Filter parameters for IMS OR faults diagnosis.	60
4.4	Filter parameters for MFPT bearing faults diagnosis.	62
4.5	Parameters of PSO algorithm.	65
4.6	Quantitative comparison of SNR.	69
5.1	Description of the Vehicle Engine Vibration Dataset.	83
5.2	2D DCNN Model.	88
5.3	Evaluation of Maximum, Minimum, Average Accuracy, and Standard Deviation Across 10 Runs for DCNN models.	90
5.4	Performance Results for 1D and 2D DCNN Models.	92
5.5	2D DCNN-LSTM Model.	97
5.6	Evaluation of Maximum, Minimum, Average Accuracy, and Standard Deviation Across 10 Runs for DCNN-LSTM models.	100
5.7	Performance Results for 1D and 2D DCNN-LSTM Models.	102
5.8	Evaluation of Maximum, Minimum, Average Accuracy, and Standard Deviation Across 10 Runs for multi-Kernel DCNN-LSTM models.	106
5.9	Performance Results for 1D and 2D multi-Kernel-DCNN-LSTM Models.	108
5.10	Comparison of Maximum, Minimum, Average Accuracy, and Standard Deviation Across 10 Runs.	110
C.1	Bearing Parameters.	129

Abstract (English)

The increasing reliance on vehicles for personal and commercial purposes underscores the critical need for efficient fault diagnosis systems to ensure reliability, reduce emissions, and maintain mechanical integrity. Among these faults, engine misfires and rotating element bearing faults present significant challenges due to their impact on performance, fuel efficiency, and environmental compliance. This thesis addresses these challenges by developing an advanced diagnostic framework that combines low-cost, high-performance MEMS accelerometers, state-of-the-art digital signal processing (DSP) techniques, and artificial intelligence (AI) models.

The ADXL1002 capacitive MEMS accelerometer was employed as the primary sensing device for its exceptional sensitivity, wide bandwidth, and cost-effectiveness. It was integrated with the BeagleBone Black controller, enabling real-time acquisition and processing of vibration signals from vehicle engines under varying operational conditions, including different RPMs, load levels, and induced misfire scenarios. Additionally, datasets of rotating element bearing vibrations from existing literature were incorporated to complement the analysis and broaden the diagnostic framework. A rigorous calibration process ensured precise and reliable measurements. The system generated a high-quality dataset of vibration signals, which forms the basis for subsequent analysis.

To extract meaningful diagnostic information from these complex and noisy signals, advanced DSP techniques were employed. These included Empirical Mode Decomposition (EMD) to isolate intrinsic mode functions, Spectral Kurtosis (SK) for transient signal analysis, and Short-Time Fourier Transform (STFT) for time-frequency representation. These methods effectively highlighted fault-specific patterns, enabling accurate identification of misfire-related characteristics. The findings were further validated using rolling bearing datasets, ensuring the robustness and generalizability of the approach.

The integration of AI into the diagnostic framework significantly enhanced fault detection capabilities. Convolutional Neural Networks (CNNs) and hybrid models combining Deep CNNs with Long Short-Term Memory (LSTM) units were developed. These models analyzed one-dimensional (1D) time-series data and two-dimensional

(2D) image representations of vibration signals. The CNNs captured spatial features in the 2D domain, while the hybrid DCNN-LSTM models effectively leveraged temporal dependencies in the 1D domain. This dual approach yielded superior diagnostic accuracy and robustness across diverse operating conditions, demonstrating the potential of AI-driven systems in real-world applications.

This research makes several noteworthy contributions: the development of a reliable real-time vibration acquisition module, the creation of a novel engine vibration dataset, the application of advanced DSP techniques for fault feature extraction, and the implementation of AI models tailored for misfire diagnosis. The methodologies proposed extend beyond automotive applications, with potential implications for fault diagnosis in industrial machinery, aerospace systems, and renewable energy technologies. By integrating low-cost sensing technologies with advanced analysis methods, this work aligns with sustainable and scalable maintenance practices, setting a benchmark for future innovations in condition monitoring and predictive maintenance.

Keywords: Engine misfire diagnosis, vibration analysis, ADXL1002 accelerometer, BeagleBone Black, digital signal processing (DSP), artificial intelligence (AI), Convolutional Neural Networks (CNN), DCNN-LSTM hybrid models, Empirical Mode Decomposition (EMD), Spectral Kurtosis (SK), Short-Time Fourier Transform (STFT), condition monitoring, predictive maintenance.

Abstract (Polish)

Rosnące uzależnienie od wykorzystywania pojazdów w celach osobistych i komercyjnych podkreśla kluczową potrzebę opracowania wydajnych systemów diagnostyki usterek w celu zapewnienia niezawodności, redukcji emisji i utrzymania integralności mechanicznej tych pojazdów. Wśród tych usterek, problemy z zapłonem silnika oraz uszkodzenia łożysk elementów obrotowych stanowią istotne wyzwania ze względu na ich wpływ na wydajność, efektywność zużycia paliwa i zgodność z przepisami środowiskowymi. Niniejsza praca podejmuje te wyzwania poprzez opracowanie zaawansowanych narzędzi diagnostycznych łączących niskokosztowe, wysokowydajne akcelerometry MEMS, najnowocześniejsze techniki cyfrowego przetwarzania sygnałów (DSP) oraz modele sztucznej inteligencji (AI).

Jako podstawowe urządzenie pomiarowe w niniejszej pracy wykorzystano pojemnościowy akcelerometr MEMS ADXL1002 ze względu na jego wyjątkową czułość, szerokie pasmo przenoszenia i koszt. Zintegrowano go z kontrolerem BeagleBone Black, co umożliwiło akwizycję i przetwarzanie sygnałów wibracyjnych z silników pojazdów w czasie rzeczywistym, w różnych warunkach eksploatacyjnych, w tym przy różnych prędkościach obrotowych, poziomach obciążenia i scenariuszach usterek. Dodatkowo, do analizy włączono zestawy danych dotyczące wibracji łożysk elementów obrotowych z istniejącej literatury, aby uzupełnić badania i rozszerzyć ramy diagnostyczne. Proces kalibracji przeprowadzony z dużą dokładnością zapewnił precyzyjne i wiarygodne pomiary. System wygenerował wysokiej jakości zestaw danych sygnałów wibracyjnych, który był podstawą do dalszych analiz.

Aby wyodrębnić istotne informacje diagnostyczne z tych złożonych i zawierających zakłócenia sygnałów, zastosowano zaawansowane techniki DSP. Obejmowały one Empiryczną Dekompozycję Modów (EMD) do izolacji funkcji modalnych, kurtozę spektralną (SK) do analizy sygnałów przejściowych oraz krótkookresową transformatę Fouriera (STFT) do reprezentacji czasowo-częstotliwościowej. Metody te skutecznie uwypukliły wzorce specyficzne dla usterek, umożliwiając dokładną identyfikację charakterystyk związanych z zapłonem. Wyniki zostały dodatkowo zweryfikowane przy użyciu zestawów danych dotyczących łożysk tocznych, co potwierdziło solidność i ogólną użyteczność podejścia.

Integracja AI z narzędziami diagnostycznymi znacząco zwiększyła zdolność wykry-

wania usterek. Zastosowano splotowe sieci neuronowe (CNN) oraz modele hybrydowe, łączące głębokie CNN z jednostkami LSTM (Long Short-Term Memory). Modele te analizowały jednowymiarowe (1D) dane czasowe i dwuwymiarowe (2D) reprezentacje sygnałów wibracyjnych w postaci obrazów. CNN wychwytywały cechy przestrzenne w domenie 2D, podczas gdy hybrydowe modele DCNN-LSTM skutecznie wykorzystywały zależności czasowe w domenie 1D. To podejście dualne zapewniło wysoką dokładność diagnostyczną i odporność w zróżnicowanych warunkach eksploatacyjnych, demonstrując potencjał systemów opartych na AI w rzeczywistych zastosowaniach.

Niniejsze badania wnoszą kilka istotnych wkładów: opracowanie niezawodnego modułu akwizycji wibracji w czasie rzeczywistym, stworzenie nowego zbioru danych wibracji silnika, zastosowanie zaawansowanych technik DSP do ekstrakcji cech usterek oraz implementację modeli AI dostosowanych do diagnostyki zapłonu. Proponowane metodologie wykraczają poza zastosowania motoryzacyjne, mając potencjalne zastosowania w diagnostyce usterek w maszynach przemysłowych, systemach lotniczych i technologiach energii odnawialnej. Dzięki integracji tanich technologii pomiarowych z zaawansowanymi metodami analizy, praca ta wpisuje się w zrównoważone i skalowalne praktyki utrzymania, ustanawiając punkt odniesienia dla przyszłych innowacji w monitorowaniu stanu i predykcyjnym utrzymaniu ruchu.

Słowa kluczowe: Diagnostyka zapłonu silnika, analiza wibracji, akcelerometr ADXL1002, BeagleBone Black, cyfrowe przetwarzanie sygnałów (DSP), sztuczna inteligencja (AI), splotowe sieci neuronowe (CNN), hybrydowe modele DCNN-LSTM, Empiryczna Dekompozycja Modów (EMD), kurtoza spektralna (SK), krótkookresowa transformata Fouriera (STFT), monitorowanie stanu, predykcyjne utrzymanie ruchu.

List of Publications

The following publications have resulted from the research conducted in this dissertation:

Journal Papers

Paper 1: Muhammad Ahsan, Dariusz Bismor, Paweł Fabiś, "A Novel Approach to Vehicle Engine Misfire Detection with ADXL1002 Accelerometer using EMD-Based Image Processing and DCNN-LSTM Model", *Vibration in Physical Systems*, Vol. , Pages: –, Year: 2025, DOI: **Accepted for Publication**

Ministry Points: 70

Paper 2: Muhammad Ahsan, Dariusz Bismor, Paweł Fabiś, "Analysis of the Vehicle Engine Misfires using Frequency-Domain Approaches at Various RPMs with ADXL1002 Accelerometer", *Archives of Acoustics*, Vol. 49, Pages: 507–516, Year: 2024, DOI: <https://doi.org/10.24425/aoa.2024.148813>

Ministry Points: 100, Q4, IF: 0.9

Paper 3: Muhammad Ahsan, Dariusz Bismor, Muhammad Arslan Manzoor, "ARL-Wavelet-BPF Optimization using PSO Algorithm for Bearing Fault Diagnosis", *Archives of Control Sciences*, Vol. 33, Pages: 589–606, Year: 2023, DOI: <https://doi.org/10.24425/acs.2023.146961>

Ministry Points: 100, Q2, IF: 1.2

Paper 4: Muhammad Ahsan, Dariusz Bismor, "Early-Stage Fault Diagnosis for Rotating Element Bearing Using Improved Harmony Search Algorithm With Different Fitness Functions", *IEEE Transactions on Instrumentation and Measurement*, Vol. 71, Pages: 1–9, Year: 2022, DOI: <https://doi.org/10.1109/TIM.2022.3192254>

Ministry Points: 100, Q1, IF: 5.6

Conference Papers

Paper 1: Muhammad Ahsan, Dariusz Bismor, Paweł Fabiś, "Comparison of ANN and CNN models for misfire detection in vehicle engine at different RPMs with

low-cost ADXL1002 accelerometer", Conference Name: 2024 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), Year: 2024, Pages: 60–65, DOI: <https://doi.org/10.23919/spa61993.2024.10715607>

Paper 2: Muhammad Ahsan, Dariusz Bismor, "Calibration of a High Sampling Frequency MEMS-Based Vibration Measurement System", Conference Name: XXI Polish Control Conference (PCC2023), Year: 2023, Pages: 294–302, DOI: https://doi.org/10.1007/978-3-031-35170-9_28

Paper 3: Muhammad Ahsan, Dariusz Bismor, "Advanced Fault Diagnosis in Rotating Machines using 2D Grayscale Images with Improved Deep Convolutional Neural Networks", Conference Name: 2023 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), Year: 2023, Pages: 77–82, DOI: <https://doi.org/10.23919/SPA59660.2023.10274428>

Paper 4: Muhammad Ahsan, Dariusz Bismor, "Early-Stage Faults Detection Using Harmony Search Algorithm and STFT-Based Spectral Kurtosis", Conference Name: Automation 2022: New Solutions and Technologies for Automation, Robotics and Measurement Techniques, Year: 2022, Pages: 75–84, DOI: https://doi.org/10.1007/978-3-031-03502-9_8

Chapter 1

Introduction

1.1 Background and Motivation

The global automotive industry is a cornerstone of modern transportation, contributing significantly to economic growth and mobility worldwide. According to production statistics reported by the Organisation Internationale des Constructeurs d'Automobiles (OICA) - International Organization of Motor Vehicle Manufacturers, the global production of vehicles has been steadily increasing, reflecting the rising demand for personal and commercial vehicles [3]. A key metric that highlights the penetration of vehicles in society is the number of vehicles per 1,000 people. Among European countries, Poland stands out with the highest vehicle ownership density, boasting 761 vehicles per 1,000 people as shown in Figure 1.1. This trend underscores the critical importance of ensuring vehicle reliability and performance, given the widespread reliance on automobiles for daily transportation.

Such high levels of vehicle ownership, particularly in countries like Poland, place considerable pressure on maintenance and diagnostic systems to ensure fleet reliability and minimize downtime. Among the various challenges faced in automotive diagnostics,

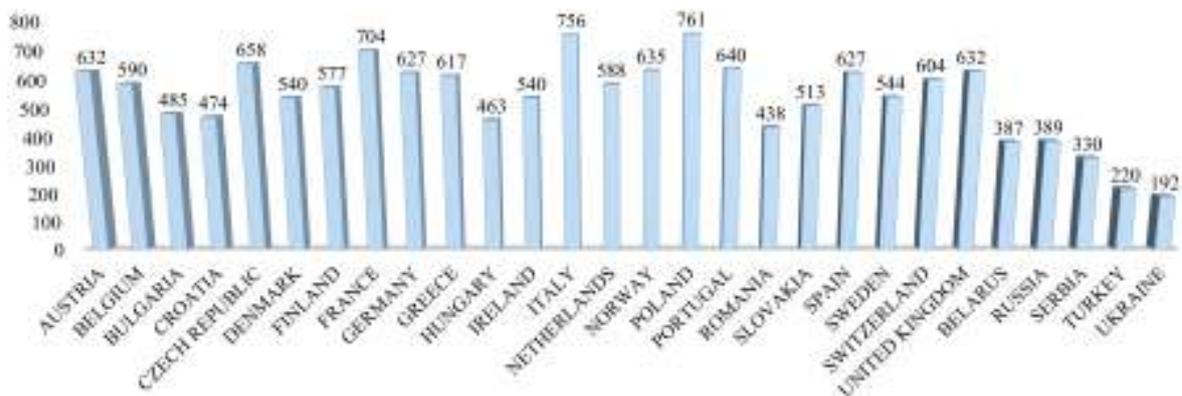


Figure 1.1: A graph showing global vehicle production per 1,000 people with a highlight on Poland's position in Europe.

the identification and diagnosis of engine misfires and different bearing faults remain critical due to their significant impact on engine performance, fuel efficiency, emissions, and overall mechanical integrity [4, 5, 6, 7, 8, 9]. Engine misfires not only affect vehicle performance but also contribute to increased emissions, which is a growing concern amidst tightening environmental regulations in Europe and globally. Similarly, bearing faults are a leading cause of mechanical failure, compromising the reliability of rotating machinery and increasing the risk of catastrophic damage if left undetected.

An engine misfire occurs when the air-fuel mixture in one or more cylinders fails to combust as intended. This incomplete combustion can lead to uneven power delivery, increased vibration, and a rise in harmful emissions. Prolonged occurrences of misfires may result in engine damage, catalytic converter failure, and a decrease in vehicle reliability [10, 11, 12, 13, 14]. Traditional methods for diagnosing engine misfires, such as on-board diagnostics (OBD-II) systems, rely on monitoring parameters like crankshaft position and exhaust gas composition. While these systems are effective to some extent, their reliance on expensive sensors and complex setups poses challenges in terms of cost, accessibility, and scalability for broader applications [15, 16, 17].

In recent years, vibration-based diagnostic techniques have gained prominence as an alternative approach for fault detection in automotive and industrial applications [18, 19, 20, 21, 22, 23, 24]. Vibration signals inherently carry rich information about the dynamic behavior of mechanical systems, including vehicle engine and different kind of bearing. By analyzing these signals, it is possible to identify anomalies associated with faults such as misfires or bearing faults. Traditional vibration analysis methods often utilize high-end piezoelectric accelerometers, which, although accurate and reliable, are costly and not suitable for widespread deployment in commercial vehicles.

The advent of Micro-Electro-Mechanical Systems (MEMS) technology has introduced a paradigm shift in the field of vibration diagnostics. MEMS accelerometers, such as the ADXL1002, offer a cost-effective solution for capturing high-frequency vibration data without compromising on sensitivity and bandwidth [25, 26, 27, 28]. These sensors are compact, energy-efficient, and capable of integrating seamlessly into embedded systems for real-time monitoring. The potential of MEMS accelerometers to replace conventional sensors in vibration analysis, combined with their affordability, makes them an attractive option for developing scalable diagnostic systems [25, 26, 28].

In addition to advancements in sensor technology, the evolution of data processing and analysis techniques has significantly enhanced the effectiveness of vibration-based diagnostics [29]. Digital Signal Processing (DSP) methods play a vital role in extracting meaningful features from raw vibration signals, enabling the identification of fault-specific characteristics [22].

Spectral analysis techniques, such as Fast Fourier Transform (FFT) and Spectral Kurtosis (SK), are widely used to identify characteristic fault frequencies in the

frequency domain. Time-frequency analysis methods, including Short-Time Fourier Transform (STFT) and Wavelet Transform (WT), offer the advantage of analyzing non-stationary signals by providing localized frequency information over time, making them particularly effective for transient fault detection. Signal decomposition approaches, such as Empirical Mode Decomposition (EMD) decompose complex vibration signals into simpler intrinsic components, facilitating the identification of localized fault signatures. Filtration techniques, including adaptive and optimized filtering, enhance fault-related features by suppressing noise and irrelevant components. Together, these methods provide a comprehensive framework for diagnosing faults in applications ranging from rotating machinery to vehicle engines, ensuring improved reliability and performance.

However, while these methods provide valuable insights, they often require domain expertise to define fault-specific features, limiting their adaptability to varying conditions and fault types. To address these limitations, the integration of Artificial Intelligence (AI) into fault diagnosis systems has emerged as a transformative development. AI methods, particularly those based on machine learning and deep learning, excel in automatically identifying patterns and features from complex datasets [19, 20, 30]. Convolutional Neural Networks (CNNs) and hybrid models like Deep Convolutional Neural Networks with Long Short-Term Memory (DCNN-LSTM) networks have shown exceptional promise in analyzing vibration signals for fault diagnosis [31, 32, 33, 34, 35]. These models leverage spatial and temporal patterns in the data, enabling accurate and robust classification of faults under diverse operating conditions.

This work is motivated by the need to develop a comprehensive diagnostic framework that combines the affordability and accessibility of MEMS accelerometers with the analytical power of DSP and AI techniques. The overarching goal is to design a system capable of diagnosing engine misfires and other faults effectively, even under real-world operating conditions. By integrating the ADXL1002 accelerometer with embedded platforms like the BeagleBone Black, this research seeks to establish a scalable solution for vehicle health monitoring.

The significance of this work extends beyond automotive diagnostics. The methodologies explored in this study have the potential to be adapted for fault detection in other mechanical systems, including industrial machinery, aerospace components, and renewable energy systems. Moreover, the approach of leveraging low-cost sensors and advanced analysis techniques aligns with the broader trend toward sustainable and efficient maintenance strategies, contributing to cost reduction, resource optimization, and enhanced reliability across various engineering domains.

In summary, this work aims to address the pressing challenges in vehicle fault diagnosis by combining cutting-edge sensor technology with state-of-the-art DSP and AI methods. By demonstrating the feasibility and effectiveness of this approach, the

research aspires to advance the state of the art in diagnostic systems, paving the way for innovations in condition monitoring and predictive maintenance.

1.2 Literature Review

1.2.1 MEMS Accelerometers and Embedded Platforms

MEMS integrate mechanical and electrical components on a single silicon chip, enabling their use in a variety of applications due to their extremely small size [28, 36, 37, 38]. The innovation, integration, and commercialization of MEMS technology have attracted significant attention, particularly in areas like vibration-based fault diagnosis in machinery and automotive systems, where commercial MEMS sensors have been extensively used. Studies have demonstrated their ability to capture high-frequency vibration data, which is essential for diagnosing faults like bearing failures and engine misfires [39, 40, 41]. Despite these advancements, research on incorporating MEMS into composite materials remains limited. The main challenge lies in their integration with composite skins, as their size still affects the mechanical properties of these materials.

The selection of an appropriate accelerometer is critical and depends on factors such as bandwidth, sensitivity, voltage noise density, zero voltage of gravity acceleration, frequency response, and dynamic range [38]. In MEMS accelerometers, achieving high performance often involves increasing the size of the sensing mass, which enhances sensitivity and measurement accuracy. However, this also reduces the maximum measurable acceleration and sensor resistance to impact [42, 43]. To overcome these trade-offs, a closed-loop circuit is often employed in these accelerometers. Additionally, MEMS accelerometers are typically tailored for specific applications, as their sensitivity and noise levels are inversely related to their ability to measure high accelerations [39, 40, 36, 37]. For instance, an accelerometer designed for high-acceleration measurements will have higher noise and broader bandwidth, whereas one optimized for precision will have lower noise and a narrower bandwidth [44, 45]. Followings are the advantages of MEMS Sensors:

- **Small size and weight:** Enables compact and portable systems.
- **Low power consumption:** Suitable for battery-powered devices.
- **High sensitivity:** Capable of detecting subtle vibrations.
- **Wide frequency range:** Covers a broad spectrum of vibration frequencies.
- **Cost-effective:** Mass production reduces costs.

Accelerometers are broadly categorized into piezoelectric, piezoresistive, capacitive, and capacitive MEMS variants, each tailored for specific applications [36, 46]. Piezoelectric accelerometers are widely used in high-frequency applications due to their reliability and low-noise characteristics; however, they are significantly more expensive

than MEMS-based alternatives and exhibit poor performance in low-frequency environments due to attenuation and phase shifts. MEMS accelerometers, available in piezoresistive and capacitive variants, offer a more cost-effective solution and are particularly well-suited for low-frequency applications.

Capacitive MEMS accelerometers are generally preferred because they exhibit less temperature sensitivity compared to their piezoresistive counterparts [46]. This is primarily due to the thermal coefficient of resistivity in piezoresistive materials, which is much higher than the thermal coefficient of capacitance in capacitive sensors. Furthermore, capacitive MEMS accelerometers have become more affordable as fabrication costs have decreased, making them the preferred choice for many applications.

Nonetheless, piezoresistive accelerometers remain advantageous in specific scenarios, such as high-frequency applications, where their low-noise properties are beneficial. Advancements in manufacturing techniques, such as deep reactive ion etching (DRIE), have also helped reduce the costs of piezoresistive sensors. These sensors are particularly useful for structural damage detection, where high-frequency local response measurements are required. In such cases, piezoresistive accelerometers are often more suitable than capacitive ones.

Capacitive MEMS accelerometers represent a specialized category within MEMS technology, designed for high-frequency applications requiring precise measurements and low noise density. While capacitive MEMS accelerometers share some similarities with traditional capacitive sensors, they are distinct due to their enhanced design and fabrication, which enable high sensitivity, broader operational ranges, and greater reliability across varying conditions. In this research, the ADXL1002 that is a capacitive MEMS accelerometer, played a crucial role, as its high-frequency response and stability under challenging conditions provided the accuracy necessary for advanced vibration analysis and condition monitoring.

Followings are the challenges and limitations of the accelerometers:

- **Sensitivity to environmental factors:** Temperature, humidity, and electromagnetic interference can affect performance.
- **Noise and drift:** Can introduce errors in measurements.
- **Limited dynamic range:** May not be suitable for high-amplitude vibrations.

The compact size and versatility of MEMS devices have significantly expanded their applications in monitoring systems, particularly in Structural Health Monitoring (SHM), Machine Condition Monitoring, and vehicle health diagnostic. MEMS accelerometers have proven indispensable in SHM, where the accurate detection of low-frequency, low-amplitude vibrations is essential for assessing structural integrity and preventing potential failures. Similarly, in Machine Condition Monitoring, these sensors enable real-time tracking of mechanical vibrations, allowing early diagnosis of faults and minimizing downtime [39, 40, 41, 47]. In this research, the use of a ca-

capitive MEMS accelerometer - the ADXL1002 - has been instrumental in diagnosing vehicle health, a critical condition requiring precise vibration analysis to identify abnormalities. Despite their vast potential in these domains, the availability of MEMS accelerometers specifically tailored for vehicle engine misfire detection remains limited in the current market, highlighting the need for further innovation and development in this area.

MEMS devices have been extensively integrated with various controllers, such as Arduino microcontrollers, FPGAs, and wireless sensor platforms, to enable real-time vibration monitoring and fault diagnosis [27, 48, 38, 49, 50]. For example, a study on a low-cost MEMS accelerometer and microphone-based condition monitoring sensor demonstrated the successful use of LoRa and Bluetooth Low Energy (BLE) radios for wireless data transmission, showcasing the potential of MEMS in industrial applications [27]. FPGA-based wireless sensor nodes have also been utilized in experimental studies for detecting unbalance faults in rotating machinery, highlighting the adaptability and efficiency of MEMS technology in such environments. These integrations underline the versatility of MEMS sensors in diverse monitoring systems, enabling precise and cost-effective solutions for condition monitoring and fault detection across various applications, including rotating machinery, vehicle systems, and industrial equipment. Followings are the interfacing challenges for MEMS sensors:

- **Noise reduction:** Filtering and signal conditioning are essential to mitigate noise.
- **Data synchronization:** Ensuring timely data transfer between sensors and controllers.
- **Power consumption:** Efficient power management is critical for battery-powered devices.

Vibration signal analysis is widely recognized as one of the most effective methods for determining the condition of machinery. It allows for real-time monitoring and offers the advantage of utilizing well-established signal processing techniques. However, it is not without challenges, as issues like noise contamination and improper sensor mounting can compromise its accuracy. Despite these limitations, vibration analysis remains the most commonly used method for fault diagnosis, with studies showing that over 82% of such techniques rely on vibration analysis [51]. Machines with moving parts inherently generate unwanted vibrations, and by analyzing these signals, decisions can be made regarding continued operation or the need for repairs.

1.2.2 Signal Processing Methods for Fault Diagnosis

Vibration signals carry rich diagnostic information about mechanical systems. To effectively extract fault-related features from these signals, various signal processing

techniques have been developed and refined. Approaches such as filter design, spectral analysis, and time-frequency decomposition are widely employed to isolate and enhance fault signatures. Signal processing methods can be classified into time-domain, frequency-domain, and time-frequency-domain.

Time-Domain Approaches

The simplest method for machine diagnosis involves analyzing vibration signals in the time domain, where signal amplitude is plotted against time. These signals, represented mathematically as:

$$x(t) = A \sin(2\pi ft + \phi),$$

where A is the amplitude, f is the frequency, ϕ is the phase, and t is the time of the signal. A simplest signal with $A = 1.0$, $f = 50$, and $\phi = \pi/4$ is shown in Figure 1.2.

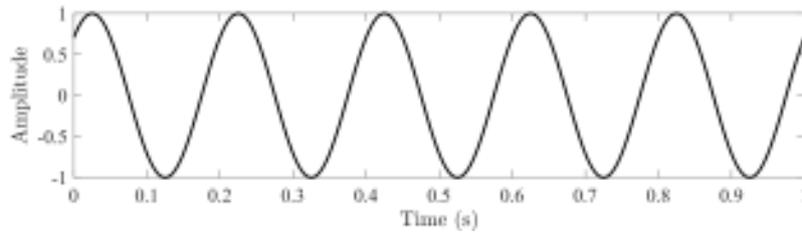


Figure 1.2: Vibration signal in the time domain: $x(t) = A \sin(2\pi ft + \phi)$ with $A = 1.0$, $f = 5.0$ Hz, and $\phi = \pi/4$.

A more complex signal includes multiple frequencies and external noises and can be given using the following equation:

$$x(t) = A_1 \sin(2\pi f_1 t) + A_2 \sin(2\pi f_2 t) + A_3 \sin(2\pi f_3 t) + A_4 \sin(2\pi f_4 t) + N,$$

We consider the following parameters for this complex signal and plot its time-domain representation in Figure 1.3; $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N is the random noise.

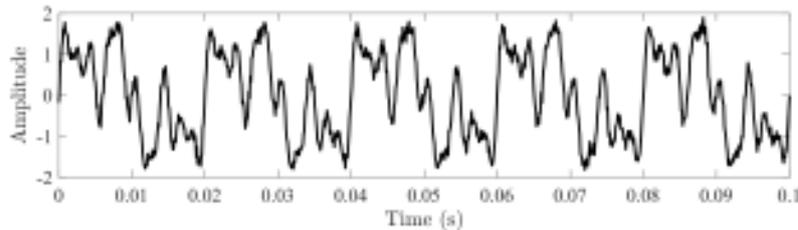


Figure 1.3: Complex vibration signal in the time domain: $x(t)$ with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N .

While visual inspection of waveforms provides useful information, it is insufficient for distinguishing specific machine failures or misfires, particularly in noisy conditions or early fault stages. To address this, statistical parameters like peak, RMS, crest factor, and kurtosis are extracted from the raw signals. These parameters help identify the most effective indicator for distinguishing healthy from defective machine states.

- **Peak Value Analysis:** The peak value represents the highest amplitude of the signal $x(t)$ during the measurement period and is calculated as:

$$\text{Peak} = \max |x(t)|.$$

When faults occur, the peak values of the vibration signal tend to fluctuate, with faults causing these values to increase. The fault's type and severity can be determined based on these amplitudes. Igba et al. [52] applied the peak value method for monitoring wind turbine gearboxes, emphasizing its effectiveness in detecting faults through variations in peak amplitudes, especially when RMS values remain unaffected by low-intensity vibrations. For a vibration signal $x(t) = A \sin(2\pi ft + \phi)$, the peak value corresponds to the amplitude A . For instance, if $A = 1.0$, the peak value is 1.0. Whereas the peak value for the complex vibration signal consists of $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, and $A_4 = 0.4$ is computed 1.9841.

- **RMS Analysis:** The RMS value represents the power content in a vibration signal and is commonly used for detecting imbalances in machinery. According to Vishwakarma et al. [53], RMS is a simple yet effective method for fault detection, particularly in steady-state conditions and single sinusoidal waveforms. Unlike peak values, RMS is less sensitive to noise, enhancing its reliability. For a pure sinusoid, the RMS value equals 0.707 times the peak amplitude. Mathematically, it is expressed as:

$$\text{RMS} = \sqrt{\frac{1}{T} \int_{T_1}^{T_2} x(t)^2 dt},$$

where T is the time duration and $x(t)$ is the signal. Figure 1.4 illustrates the vibration signal and the RMS value as a horizontal reference line.

However, Igba et al. [52] noted that RMS is less sensitive to isolated peaks and

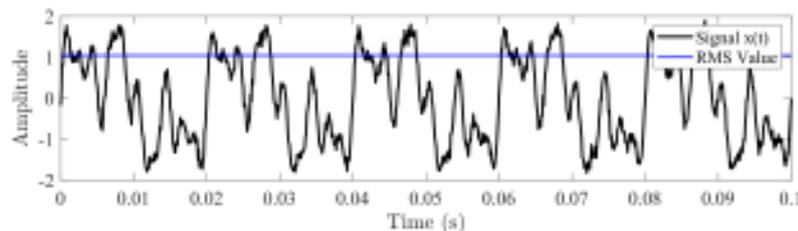


Figure 1.4: RMS of the complex vibration signal: $x(t)$ with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N .

short, low-intensity vibrations, limiting its effectiveness in detecting early-stage failures in bearings and gears. Despite these limitations, RMS has been applied in several diagnostics, such as by Bartelmus et al. [54] for gearbox fault detection and Sheldon et al. [55] for wind turbine monitoring. While it is less effective for early failure detection, Krishnakumari et al. [56] combined RMS with fuzzy logic for spur gear diagnostics, achieving 95% accuracy. RMS remains a valuable parameter in machine condition monitoring.

- **Crest Factor Analysis:** The crest factor, defined as the ratio of a signal's peak value to its RMS value, is given by:

$$\text{Crest Factor} = \frac{\text{Peak Value}}{\text{RMS Value}}$$

For the given signal $x(t) = A \sin(2\pi ft + \phi)$, the peak value is 1.0, the RMS value is 0.707, and the crest factor is calculated as:

$$\text{Crest Factor} = \frac{1.0}{0.707} = 1.414$$

For a pure sine wave, the crest factor is approximately 1.414, while for random noise with a normal distribution, it is around 3. However, for the complex signal constructed above has the crest factor 1.9016. Unlike peak or RMS values, the crest factor remains unaffected by rotational speed, making it suitable for analyzing signals at varying speeds. However, it is only reliable in cases of significant impulsiveness.

Shrivastava and Wadhwani [57] found that while the crest factor, along with other time-domain features, could detect electrical machine faults, it struggled to differentiate between healthy bearings and bearings with specific defects. Aiswarya et al. [58] used the crest factor and other features to diagnose faults in a rocket engine turbo pump. Combined with SVM, their method achieved 100% fault classification accuracy.

- **Kurtosis Analysis:** Kurtosis is a statistical feature of the probability density function to determine the strength of transient impulses of the non-stationary signal. In statistics, kurtosis can be defined as the fourth standardized moment [59, 60]. In machine condition monitoring, kurtosis is often preferred over the crest factor, as it provides a more sensitive detection of faults. While crest factor meters are widely available and affordable, kurtosis meters can offer greater fault sensitivity. For instance, Fu et al. [61] incorporated kurtosis into an unsupervised AI method for diagnosing rolling bearing faults, showing its effectiveness in detecting even slight issues. Similarly, Runesson [62] used kurtosis along with RMS values to

monitor the condition of a mechanical press, finding that although kurtosis alone may not be fully reliable, it still provides valuable insights into the gearbox condition. For a signal x with N number of samples and \bar{x} sample means, the kurtosis is given as follows:

$$K = \frac{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^4}{\left(\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2 \right)^2}. \quad (1.1)$$

The value of the kurtosis increases with an increase in the transient impulses. For bearing vibration analysis, we also use analytic signal \tilde{x}_i to express the kurtosis as follows:

$$K = \frac{E(|\tilde{x}_i|^4)}{\left(E(|\tilde{x}_i|^2) \right)^2}; \quad (1.2)$$

where $E(\cdot)$ is the expectation operator. The analytical signal \tilde{x}_i can be derived from the time-domain zero-mean filtered signal x_i as follows:

$$\tilde{x}_i = x_i + j \cdot H(x_i); \quad (1.3)$$

where $H(\cdot)$ is the Hilbert Transform function.

When faults occur in a rotating machine such as a bearing, high-frequency transient impulses generated periodically raise the value of the kurtosis. This phenomenon can be used to determine transient faults and diagnose bearing faults. However, a simple kurtosis calculation cannot distinguish between noise and fault impulses, and it isn't easy to characterize the resonant band of the filtered signal.

Frequency-Domain Approaches

Building on the foundation of time-domain analysis, real-world vibration signals can often be decomposed into distinct sine waves. In the frequency domain, each sine wave is represented as a vertical line, with its height corresponding to amplitude and its position indicating frequency. By plotting amplitude against frequency, resonant frequency components become more apparent, making frequency domain methods highly effective for detecting machine faults. This clarity is a significant advantage over time-domain analysis, as certain signal characteristics that remain hidden in the time domain can be easily observed in the frequency domain.

- **FFT Analysis:** Fourier Transform (FT) is a mathematical tool used to convert a time-domain signal $f(t)$ into its frequency-domain representation $F(\omega)$, where

ω represents frequency. The FT is expressed as:

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt$$

The inverse FT (IFT) allows the signal to be converted back to the time domain:

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega)e^{i\omega t} d\omega$$

For discretized time signals, the FFT is an efficient algorithm to compute the FT. Utilizing the FFT in the misfire and engine analysis involves decomposing vibration signals into their fundamental frequency components, unveiling spectral patterns inherent within the signals [63].

The FFT algorithm computes the Discrete FT (DFT) efficiently. Mathematically, the DFT of a discrete-time function $f(n)$ can be represented as follows [63]:

$$F(k) = \sum_{n=0}^{N-1} f(n)e^{-i2\pi kn/N}; \quad (1.4)$$

In the above equation, $F(k)$ denotes the complex values within the frequency domain at a specific index k . This complex value represents the transformed signal's amplitude and phase at a particular frequency component. On the other hand, $f(n)$ signifies the discrete-time signal in the time domain at a distinct index n . The variable N represents the total count of samples constituting the time-domain signal. The term $e^{-i2\pi kn/N}$ is a complex exponential expression encapsulating the phase shift and frequency of individual components within the signal.

Despite its efficiency, FFT has limitations, including minor loss of time information during domain conversion and difficulty in analyzing transient features. While FFT can identify faults, it does not effectively assess fault severity. Combining time-domain analysis with FFT is often used for diagnosing low-speed

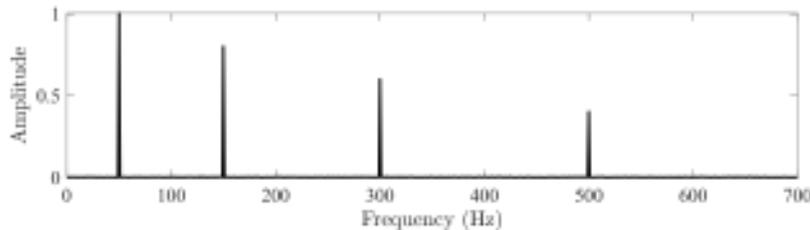


Figure 1.5: FFT of the complex vibration signal: $x(t)$ with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N .

machinery to improve accuracy, though its performance can depend on fault magnitude affecting the carrier frequency.

Consider the same complex vibration signal as shown above. The frequency domain representation of this signal reveals the frequency components distinctly as shown in Figure 1.5.

- **Envelope Analysis:** Envelope analysis, also referred to as amplitude demodulation or demodulated resonance analysis, was first introduced by Mechanical Technology Inc. [64]. This method isolates low-frequency signals from background noise through bandpass filtering and demodulation, extracting the signal envelope that often contains critical diagnostic information [53, 65]. Figure 1.6 illustrates the same complex vibration signal as developed above and its envelope.

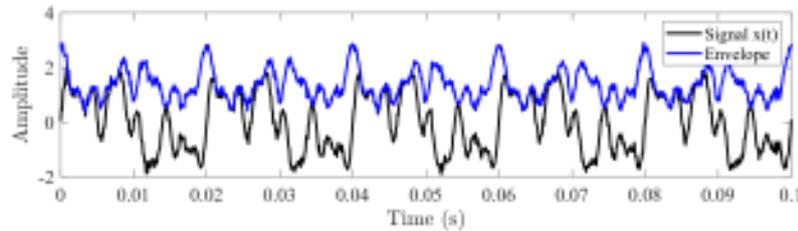


Figure 1.6: Complex vibration signal and its envelope with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N .

However, its effectiveness depends on accurately selecting the optimal frequency band for filtering, requiring sharp filters and precise specifications. Noise interference in bearing failure signals can complicate envelope analysis, but the squared envelope method has addressed this issue, proving particularly effective for cyclostationary signals [66]. Rubini and Meneghetti [67] compared envelope analysis to Wavelet Transform (WT) for detecting faults in ball bearings. They found that while envelope analysis failed after prolonged use (30 minutes or 48,000 cycles), WT remained effective.

Envelope analysis has also been applied by Widodo et al. [68] to preprocess vibration signals of low-speed bearings, identifying characteristic frequencies. However, when paired with SVM for fault recognition, it performed worse than acoustic emission (AE) analysis. Similarly, Leite et al. [69] successfully used envelope analysis to detect induction motor bearing faults without relying on model-specific information.

- **Spectrum Analysis:** Spectrum analysis is closely associated with FFT, as FFT is commonly used to transform signals from the time to the frequency domain. For effective comparison, spectrum analysis is typically performed on a logarithmic amplitude scale (dB), as changes on this scale can highlight variations in vibration states. However, small fluctuations in a machine's rotational speed can

pose challenges. This method is particularly effective in identifying faults that cause significant changes in vibration signatures over short durations.

In cases where faults cause amplitude modulation in the vibration signal, envelope spectrum analysis is highly effective. This technique extracts the modulation characteristics, which can highlight repetitive patterns or fluctuations often associated with misfire faults or bearing faults. By focusing on these modulations, envelope analysis can help identify specific engine irregularities.

The computation of the Envelope Spectrum (ES) stands as an essential method to discern nuanced variations induced by both load and misfire events. This technique entails the extraction of the signal's envelope, thereby highlighting alterations in the overall vibration behavior attributed to changes in load or misfires. This approach furnishes accurate diagnostic insights into the operational state of the engine, pinpointing specific modifications in the vibration patterns influenced by varying loads or misfire occurrences [70].

The ES represents the amplitude of specific frequency components within a signal. The Hilbert transform is used to compute the complex envelope of the signal. Once the envelope is obtained, the FFT is applied to it and to focus on the positive frequencies, only the first half of the FFT result is retained. The ES is then computed by taking the squared magnitude of the FFT of the envelope and normalizing it by the length of the signal, N . Finally, the Envelope Spectrum is plotted as a function of frequency. The resulting plot will illustrate the ES, which represents the frequency content of the signal's envelope. This spectrum provides insight into how the amplitude modulations of the signal are distributed across different frequencies as illustrated in Figure 1.7. While spectrum analysis is widely studied, it remains a complex technique requiring expertise for effective application. Unlike cepstrum analysis, spectrum analysis does not provide time localization of frequency components. Ciabattini et al. [71] introduced a statistical spectrum analysis (SSA) method, which transforms FFT-derived spectra into statistical spectral images for fault diagnosis in rotating machines.

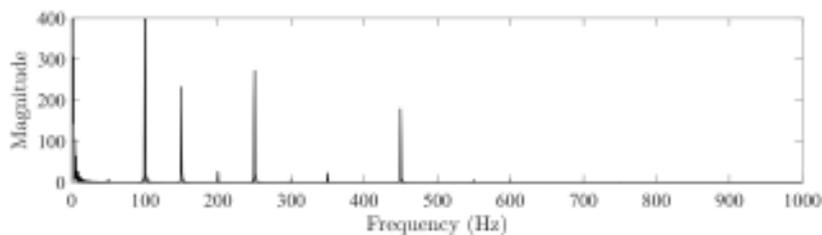


Figure 1.7: ES of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N .

Time-Frequency-Domain Approaches

Time-frequency domain analysis combines the principles of time and frequency domains, allowing simultaneous determination of signal frequency components and their time-varying characteristics. Unlike traditional time and frequency domain methods, which rely on the assumption of signal stationarity and cannot effectively capture localized features, time-frequency domain approaches are specifically suited for analyzing non-stationary signals.

Methods such as WT, Hilbert-Huang Transform (HHT), Wigner-Ville Distribution (WVD), STFT, SK, and Power Spectral Density (PSD) are commonly employed in this analysis. Each method has unique advantages and limitations, making them suitable for specific applications.

- **WT Analysis:** WT is a linear transformation that breaks down a time-domain signal into localized wavelets, which are functions of time with specific frequency content [72]. Unlike traditional Fourier transforms that rely on sinusoidal basis functions, WT employs wavelets, and its effectiveness depends on selecting a wavelet basis suited to the signal structure [73]. This choice is critical to ensure accurate diagnosis. Compared to STFT, WT provides superior time resolution at higher frequencies and is particularly advantageous for analyzing nonstationary and transient signals. Zou and Chen [74] highlighted its sensitivity to stiffness variations over Wigner-Ville Distribution (WVD).

WT is categorized into Discrete WT (DWT) and Continuous WT (CWT). DWT uses powers of two as scaling factors, typically implemented via lowpass and high-pass filters. CWT, in contrast, employs arbitrary scaling factors, often through convolution. The continuous wavelet transform $W(\alpha, \beta)$ of a signal $x(t)$ is given by [75]:

$$W(\alpha, \beta) = |\alpha|^{1/2} \int_{-\infty}^{+\infty} x(t) \phi^* \left(\frac{t - \beta}{\alpha} \right) dt, \quad (1.5)$$

where α and β represent scale and shift parameters, respectively; $\phi(\cdot)$ is wavelet function; and $*$ shows the conjugate. In the frequency domain, it can be written as follows:

$$W(\alpha, \beta) = |\alpha|^{1/2} IFT[X(f)\psi^*(\alpha f)], \quad (1.6)$$

where $X(f)$ and $\psi(f)$ are the Fourier transform of the signal and wavelet function, respectively; and IFT represents the inverse Fourier transform.

The wavelet function in Equation (1.5) is composed of a sine wave with a Gaussian function. This sine wave could be real or complex. While both are standard WT methods, they may fail to produce sparse representations, leading to low-frequency resolution for high-frequency components and poor time localization for low-frequency components. To address these issues, the Wavelet Packet

Transform (WPT) extends CWT by further decomposing high-frequency details, enhancing frequency resolution for nonstationary signals.

- **EMD Analysis:** EMD is a time-frequency analysis technique that decomposes a signal into Intrinsic Mode Functions (IMFs). These IMFs represent different oscillatory modes within the data, allowing for a detailed examination of non-stationary signals. In misfire diagnosis, the first few IMFs often reveal high-frequency components indicative of transient events, such as misfires. Figure 1.8 illustrates the EMD of the constructed complex vibration signal.

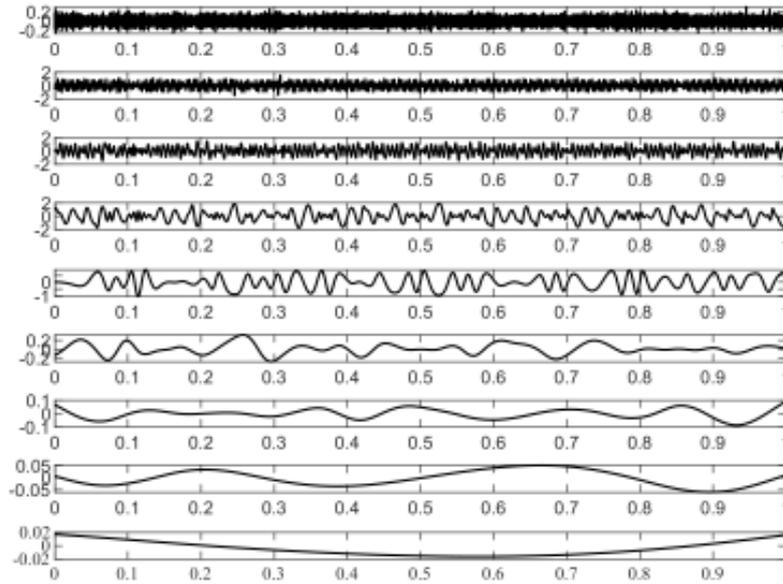


Figure 1.8: EMD of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N .

- **HHT Analysis:** HHT is used to analyze stationary, nonstationary, and transient signals. The HHT combines EMD with the Hilbert transform, allowing for the generation of a Hilbert spectrum that aids in fault diagnosis of running machines [73]. The EMD method decomposes a complex signal into a set of IMFs, which, when combined, can reconstruct the original signal, expressed as:

$$x(t) = \int_{-\infty}^{\infty} c_i(t) + r_n(t) dt \quad (7)$$

where $c_i(t)$ represents the i -th IMF and $r_n(t)$ is the residual signal that captures the signal's low-frequency trend [73]. The HHT offers advantages such as low computational time and the absence of convolution operations [76]. However, EMD has limitations, including the misinterpretation of IMFs generated at low frequencies, as well as the inability to separate low-frequency components. To

address these issues, the ensemble EMD (EEMD) method was introduced by adding Gaussian white noise to EMD [73, 77].

Peng et al. [78] enhanced the HHT by integrating the WPT to decompose vibration signals into narrowband components. Their method demonstrated superior time-frequency resolution compared to traditional wavelet-based scalograms. Wu et al. [79] applied the HHT for diagnosing looseness faults in rotating machinery, successfully detecting faults in various machine components. Osman and Wang [80] introduced a normalized HHT (NHHT) technique to address the challenge of selecting the correct IMF components, particularly for bearing health monitoring. However, the method was limited to narrowband signal processing.

- **STFT Analysis:** STFT, introduced by Gabor in 1946 [81], is a powerful tool used for analyzing signals in the time-frequency domain, particularly useful for non-stationary signals, where frequency content changes over time. Traditional FFT methods provide frequency information but lack temporal localization, which limits their effectiveness in capturing transient or time-varying characteristics in signals such as vibration or audio. STFT overcomes this limitation by applying the Fourier Transform to short segments of the signal, each weighted by a window function, allowing it to track frequency content changes over time.

For any discrete stochastic process $x(n)$ with given analysis window $w(n)$ of length N , the STFT of $x(n)$ is defined as follow [82, 83]:

$$STFT(k, f) = \sum_{n=1}^N x(n)w(n - k)e^{-j2\pi n f}; \quad (1.7)$$

where $w(n - k)$ is the short-time window with the shift k . The window function's length and shape determine the trade-off between time and frequency resolution; shorter windows provide better temporal resolution but lower frequency resolution, and vice versa.

In STFT, the signal $x(n)$ is segmented into overlapping windows, each transformed into the frequency domain to provide a two-dimensional representation, $STFT(k, f)$, capturing both time and frequency information. This representation is particularly useful for signals that exhibit transient events or localized frequency components, as it provides a continuous view of how the frequency content evolves over time. Figure 1.9 illustrates the STFT of the constructed complex vibration signal.

Due to its dual resolution in time and frequency, STFT is widely used in applications like speech processing, vibration analysis, and biomedical signal processing. In vibration analysis, it is instrumental in identifying fault-related frequencies in rotating machinery and other mechanical systems. By capturing the evolution of frequency components, STFT allows for the detection of anomalies and localized

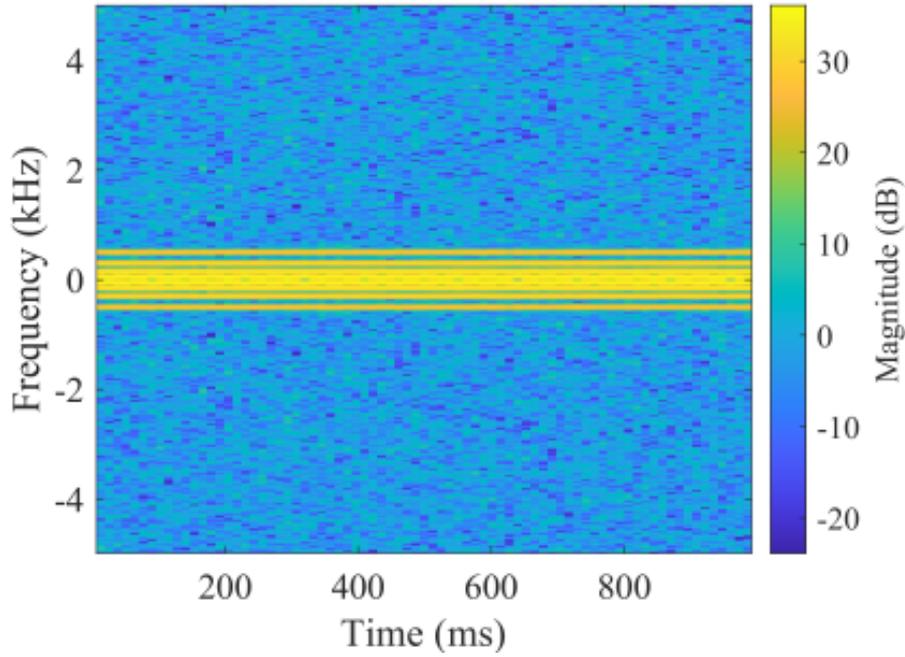


Figure 1.9: STFT of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N .

transients, providing a foundation for further analysis through SK and other advanced signal processing methods.

To address cross-term effects, Burriel-Valencia et al. [84] applied STFT for fault detection in induction machines by filtering spectra in relevant frequency bands, thus reducing computational time and memory usage. STFT has also been successfully used in fault diagnosis of induction motors [85, 86], and rolling element bearings [87, 70].

- **SK Analysis:** SK describes the kurtosis over each frequency band and allows us to discover hidden transients and the frequency band in which these transients occur. We can calculate the SK value for the non-stationary signal in the presence of noise as follows:

$$K_{x+n}(f) = \frac{K_x(f)}{[1 + \rho(f)]^2}; \quad (1.8)$$

where $\rho(f)$ is the function of frequency representing noise-to-signal ratio and $K_x(f)$ is the SK without noise. The latter can be given as the fourth-order normalized cumulant:

$$K_x(f) = \frac{\lim_{x \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N |H(n, f)|^4}{\left(\lim_{x \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N |H(n, f)|^2 \right)^2} - 2; \quad (1.9)$$

where $H(n, f)$ is the complex envelope of $x(n)$. It can also be represented as follows:

$$K_x(f) = \frac{\langle H^4(t, f) \rangle}{\langle H^2(t, f) \rangle^2} - 2, \quad (1.10)$$

where $\langle \cdot \rangle$ represents the time averaging operator, and $H(t, f)$ is the time-frequency envelope spectrum of the vibration signal $x(t)$.

The relationship between the signal $x(n)$ and complex envelop $H(n, f)$ is given by the following equation:

$$x(n) = \int_{+1/2}^{-1/2} H(n, f) e^{j2\pi f n} dZ_n(f). \quad (1.11)$$

- **STFT-SK Analysis:** By combining STFT with SK (STFT-SK), it becomes possible to identify not only the time and frequency of faults but also the intensity of fault-related components, enhancing diagnostic precision. In Equation (1.9), the complex envelope $H(n, f)$ is the function of time and frequency. The stochastic process $x(n)$ in equation (1.11) is characterised by a double stochasticity, both in $H(n, f)$ and $dZ_n(f)$. STFT is an efficient tool that can estimate better results and in the proposed algorithm, $H(n, f)$ is replaced with STFT. For any stochastic process $x(n)$ with given analysis window $w(n)$ of length N , the STFT of $x(n)$ is defined as follow [59, 82, 83]:

$$STFT(k, f) = \sum_{n=1}^N x(n) w(n - k) e^{-j2\pi n f}; \quad (1.12)$$

where $w(n - k)$ is the short-time window with the shift k . The STFT based SK is given by the following equation [59]:

$$SK_{STFT} = \frac{\lim_{x \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N |STFT|^4}{\left(\lim_{x \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N |STFT|^2 \right)^2} - 2. \quad (1.13)$$

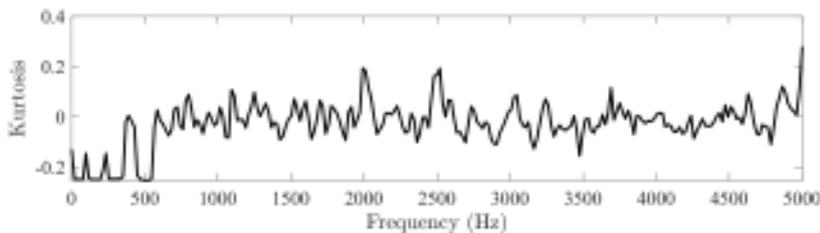


Figure 1.10: STFT-based-SK of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N .

Figure illustrates the STFT-based-SK of the constructed complex vibration signal.

- **PSD Analysis:** PSD is used to measure the amplitude of oscillatory signals in time series data and assess the energy of frequencies, which is helpful for further analysis. The one-sided PSD, expressed in $(m/s^2)^2/Hz$, is calculated from the complex spectrum as:

$$PSD(f) = \frac{2|X(f)|^2}{t_2 - t_1}$$

where $t_2 - t_1$ is the time range, and $X(f)$ is the complex spectrum of the vibration $x(t)$, measured in units of $(m/s^2/Hz)$. Alternatively, PSD can be computed in the frequency domain by using the FFT of the vibration signal, as given by the formula:

$$PSD = \frac{(Grms)^2}{f}$$

where $Grms$ is the root mean square of the acceleration at frequency f . Figure illustrates the PSD of the constructed complex signal.

PSD is useful for analyzing faulty frequency bands without facing slip variation issues and does not require focusing on a single harmonic. It is computationally efficient and can be directly computed through FFT or by converting the auto-correlation function. PSD has been employed by Yi et al. for fault diagnosis in rolling bearing [88].

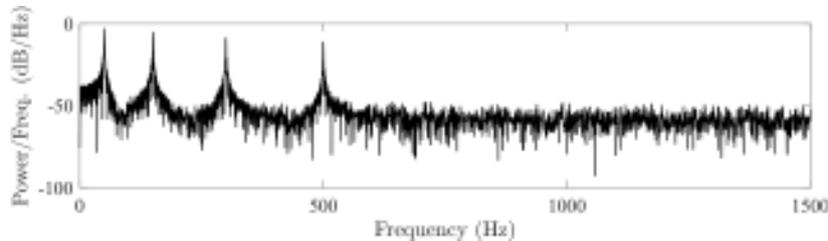


Figure 1.11: PSD of the complex vibration signal with $A_1 = 1.0$, $A_2 = 0.8$, $A_3 = 0.6$, $A_4 = 0.4$, $f_1 = 50$ Hz, $f_2 = 150$ Hz, $f_3 = 300$ Hz, $f_4 = 500$ Hz, and N .

1.2.3 Machine Learning Methods for Fault Diagnosis

AI has become increasingly prominent in fault diagnosis, offering a suite of methods that excel in handling complex and large-scale datasets, adapting to dynamic environments, and automating diagnostic tasks. AI-based techniques, particularly Artificial Neural Networks (ANN) and CNN, have been applied across various domains to effectively diagnose faults in mechanical systems by identifying patterns and anomalies within signal data. In the context of vehicle engine misfire diagnosis, AI techniques offer substantial advantages over traditional methods by eliminating the need for manual feature extraction, instead automatically identifying fault-relevant features directly

from vibration signals. This section reviews key AI methods, including ANN and CNN, as well as hybrid architectures, which have been utilized to enhance fault diagnosis in vehicle engines.

ANN Models

ANN are one of the foundational architectures in AI, modeled loosely after the way neurons in the human brain process information. In fault diagnosis, ANNs have demonstrated a capacity for detecting complex, non-linear relationships within large datasets, making them well-suited for applications where underlying patterns are not easily distinguishable through conventional signal processing methods.

For engine misfire detection, ANNs can process vibration signals and classify them according to the presence or absence of fault conditions. Their use in fault diagnosis typically involves feeding a preprocessed vibration dataset into the ANN model, allowing it to learn patterns specific to various fault types, such as misfire events. However, while ANNs have shown promising results, they have certain limitations when dealing with raw vibration signals, especially when identifying spatial or frequency-based features that are often significant in vibration data analysis.

CNN Models

CNNs, originally developed for image recognition tasks, have also been highly effective in fault diagnosis, including in the domain of vibration-based analysis. CNNs are designed to automatically extract and learn spatial patterns from data through multiple layers of convolution and pooling, capturing hierarchical features that help in differentiating between normal and faulty conditions.

In this research, CNNs were used to analyze both one-dimensional (1D) and two-dimensional (2D) representations of vibration signals to detect misfire events in vehicle engines. For 1D CNNs, raw vibration data is fed directly into the model, allowing it to learn from the time-domain characteristics of the signal. The advantage of 1D CNNs in this context is that they require minimal preprocessing and can efficiently capture patterns and anomalies specific to misfire without converting the signals into alternative forms. By processing 1D vibration data directly, 1D CNN models focus on temporal features inherent in the vibration data, providing a straightforward and computationally efficient approach to misfire detection.

For image-based diagnosis, vibration signals were transformed into 2D grayscale representations and then analyzed by CNN models. This approach leverages the spatial feature extraction capabilities of CNNs in a way that allows the model to capture both temporal and frequency-based patterns indicative of misfire events. The image-based input format enables CNNs to learn from visual representations of vibration data,

effectively identifying fault-related features that may not be as apparent in raw time-series data. By combining these 1D and 2D approaches, this study sought to leverage the strengths of CNN models in capturing diverse features present in both raw and processed representations of vibration signals.

Deep CNN (DCNN) and Hybrid DCNN-LSTM Models

The demand for higher accuracy and robustness in fault diagnosis has led to the development of advanced AI architectures, such as DCNN and hybrid models that integrate DCNN with LSTM networks. DCNNs are a deeper variant of CNNs, typically consisting of multiple convolutional layers that allow for the extraction of more complex and high-level features from the data. For fault diagnosis in vehicle engines, DCNNs have proven beneficial, particularly in learning subtle features associated with misfire events from vibration data.

In this study, a DCNN was implemented to process both 1D vibration data and 2D grayscale images, allowing the model to take advantage of the unique features each data format offers. The deeper architecture of the DCNN provides an enhanced ability to capture intricate fault-related patterns that may not be apparent in shallow architectures or simpler models. The combination of DCNN and 2D images aids in capturing spatial and frequency patterns, while the DCNN's capability to handle 1D data allows it to focus on temporal variations in the vibration signal.

To address the temporal dependencies within the vibration signals, a hybrid model was constructed by integrating DCNN with LSTM units. LSTM networks are a type of recurrent neural network (RNN) designed to retain information over time, making them suitable for sequential data such as vibration signals, where patterns can evolve over multiple time steps. The DCNN-LSTM model leverages the feature extraction power of DCNN with the sequence learning capability of LSTM, thereby capturing both spatial and temporal dependencies. This hybrid approach is particularly beneficial in diagnosing intermittent faults, such as misfire, where temporal patterns play a critical role in identifying fault occurrences accurately.

1.3 Research Objectives

The primary objective of this research is to develop a comprehensive framework for diagnosing engine misfires using vibration signal analysis. To achieve this, the first goal was to design and implement a reliable vibration data acquisition module. The ADXL1002 accelerometer was chosen due to its high sensitivity, wide frequency range, low noise density, and low price, making it ideal for capturing precise vibration signals from vehicle engines. The accelerometer was interfaced with the BeagleBone Black

controller, which served as the data acquisition platform. The integration process included sensor setup, signal conditioning to minimize noise, and the development of software capable of real-time acquisition, processing, and storage of vibration signals. Once the module was successfully prepared, vibration data were recorded from the vehicle engine under various conditions, including different RPMs, load levels, and intentional misfire scenarios. This effort also aimed to generate a high-quality dataset of engine vibration signals that can serve as a benchmark for further analysis. In addition to utilizing a dataset of vehicle engine misfire signals, this research also incorporated bearing datasets featuring various fault conditions to develop and evaluate DSP and ML models. These bearing datasets, which have been extensively documented and are readily available in the literature, provided a robust foundation for the analysis and validation of the proposed methodologies.

The second objective focused on applying DSP techniques to analyze vibration signals for fault diagnosis. Vibration signals, being inherently noisy and complex, require advanced preprocessing methods to enhance signal quality. Initially, DSP methods such as kurtosis, SK, STFT, and CWT were used to design optimized filters. These filters were validated on the bearing datasets to demonstrate their effectiveness in isolating fault-related features. Subsequently, EMD was applied to the vehicle engine vibration dataset to extract fault-specific features, particularly those related to misfire events. These techniques enabled the identification of unique vibration patterns and frequency components associated with both rolling bearing faults and engine misfire conditions, providing a strong foundation for accurate and reliable fault detection.

The final objective was to utilize ML methods to improve the accuracy and robustness of engine misfire diagnosis. ML techniques, particularly deep learning models, were employed to automatically learn complex patterns and relationships within the vibration data. Both 1D signals and their 2D representations (e.g., grayscale images) were used as inputs for the models. Specifically, architectures such as DCNN, and DCNN-LSTM hybrid models were implemented to classify and detect misfire conditions. The models were trained and evaluated using the prepared vehicle engine dataset, with performance metrics such as accuracy, precision, and recall used to assess their effectiveness. By comparing the results of ML methods, this research aimed to establish a robust and reliable diagnostic system for vehicle engine misfire detection under varying operating conditions.

1.4 Contributions

- **Development of a Real-Time Vibration Data Acquisition Module:** Successfully interfaced the low-cost ADXL1002 accelerometer with the BeagleBone Black controller to create a robust module capable of real-time vibration signal

acquisition under varying engine conditions, including RPM, load, and misfire scenarios.

- **Creation of a New Engine Vibration Dataset:** Recorded a high-quality dataset of vehicle engine vibration signals under diverse operating conditions, including varying RPMs, load levels, and intentional misfire events, providing a valuable resource for fault diagnosis research.
- **Design and Validation of DSP-Based Filters:** Applied advanced DSP techniques like kurtosis, SK, STFT, and CWT to optimize filters for fault detection using vibration datasets of rolling bearing.
- **Application of DSP Techniques for Misfire Diagnosis:** Applied advanced DSP techniques, including EMD to extract the first IMF and DFT to identify misfire-related frequency components in the engine vibration signals.
- **Implementation of ML Models for Misfire Diagnosis:** Developed and applied DCNN and DCNN-LSTM models to diagnose engine misfire conditions using both 1D vibration signals and 2D grayscale image representations.

1.5 Structure of the Dissertation

The dissertation is organized as follows:

- **Chapter 2:** Discusses the interfacing of the ADXL1002 accelerometer with the BeagleBone Black and its calibration in a controlled laboratory environment to ensure accurate measurements.
- **Chapter 3:** Presents datasets collected from vehicle engine misfire scenarios at various RPMs and includes benchmark bearing datasets from the literature.
- **Chapter 4:** Explores DSP techniques such as filter design, FFT, and EMD for fault diagnosis in both bearings and vehicle engine vibrations.
- **Chapter 5:** Investigates AI methods for fault diagnosis, focusing on DCNN, DCNN-LSTM, and multi-Kernel DCNN-LSTM models using 1D and 2D representations of vibration signals.
- **Chapter 6:** Provides the conclusion of this research, summarizing the key findings from the previous chapters and discussing their significance in health diagnosis using vibration signals. It highlights the major contributions, limitations, and possible directions for future work to enhance fault detection methodologies.

Chapter 2

Interfacing and Calibration of the ADXL1002 Accelerometer with BeagleBone Black

2.1 Introduction

In this chapter, we explore the interfacing and calibration of the ADXL1002 accelerometer with the BeagleBone Black, a single-board computer widely used for embedded systems applications [2, 89, 90]. This chapter is structured into two main sections: the first section focuses on the interfacing of the ADXL1002 accelerometer with the BeagleBone Black, while the second section delves into the calibration of the ADXL1002 accelerometer.

The first section begins by describing the key components involved in the interfacing process, namely the ADXL1002 accelerometer and the BeagleBone Black. Detailed descriptions of the main block diagram of the interfacing circuit and its various modules, such as the power supply module, the filter and voltage divider circuit, and the LED and switch circuits, are provided. This section aims to give a comprehensive understanding of how the ADXL1002 accelerometer is interfaced to the BeagleBone Black and the role of each component in ensuring accurate data acquisition.

In the second section, we discuss the calibration of the ADXL1002 accelerometer [90]. Calibration is crucial to ensure the accuracy and reliability of the accelerometer's readings [91, 92]. This section includes the description of a vibration laboratory setup designed to provide a controlled environment for calibration purposes. The setup allows for the generation of different frequency signals, which are then recorded using the ADXL1002 accelerometer. The calibration process involved adjusting the sensor such that the recorded frequencies matched the reference frequencies as closely as possible, thereby minimizing any errors between the two [90]. Additionally, the sensitivity

characteristics of the accelerometer at various frequencies were examined to ensure its performance across a range of conditions.

The results of the calibration process, including the accuracy of frequency matching and sensitivity analysis, have been previously published in a conference proceeding [90], demonstrating the validity and reliability of the methods used. These results underscore the importance of proper interfacing and calibration in achieving precise and dependable measurements with the ADXL1002 accelerometer.

2.2 Interfacing

This section provides a detailed description of the interfacing process involving the ADXL1002 accelerometer, the BeagleBone Black, and the developed modules. The developed modules including the power supply module for stable operation, the filter and voltage divider circuit for signal conditioning, and the LED and switch circuits for user interaction, ensuring a comprehensive understanding of the system's design and functionality.

2.2.1 ADXL1002 Accelerometer

The ADXL1002 is a high-frequency, low-noise, single-axis microelectromechanical systems (MEMS) accelerometer that provides an analog output proportional to mechanical vibration [2, 27]. The accelerometer is integrated into the EVAL-ADXL1002Z evaluation board, which is specifically designed for mounting onto mechanical shakers. This evaluation board is constructed with an extra-thick printed circuit board (PCB) measuring 0.8 inches square and includes screw holes for secure attachment to a shaker block, ensuring accurate vibration analysis (see Figure 2.1). Figure 2.1 illustrates the EVAL-ADXL1002Z evaluation board alongside the function block diagram of the ADXL1002 accelerometer. The block diagram details the internal architecture of the accelerometer, highlighting key components such as the sensor element, signal conditioning circuitry, and output stage, which together enable precise detection and measurement of mechanical vibrations [2, 27].

The ADXL1002 is engineered for high-bandwidth applications, capable of measuring g-forces up to 50 g, making it suitable for vibration analysis systems used in monitoring and diagnosing machinery health. The device is shock-proof up to 10,000 g, and guarantees long-term durability even in harsh environments. ADXL1002 operates within a supply voltage range of 3.3 V to 5.25 V. It comes in an ultra-small LFCSP package that has an operational temperature range of -40°C to $+125^{\circ}\text{C}$ [2].

The accelerometer's low noise and wide frequency bandwidth enable it to detect vibration patterns from small moving parts, such as internal bearings. The high g range

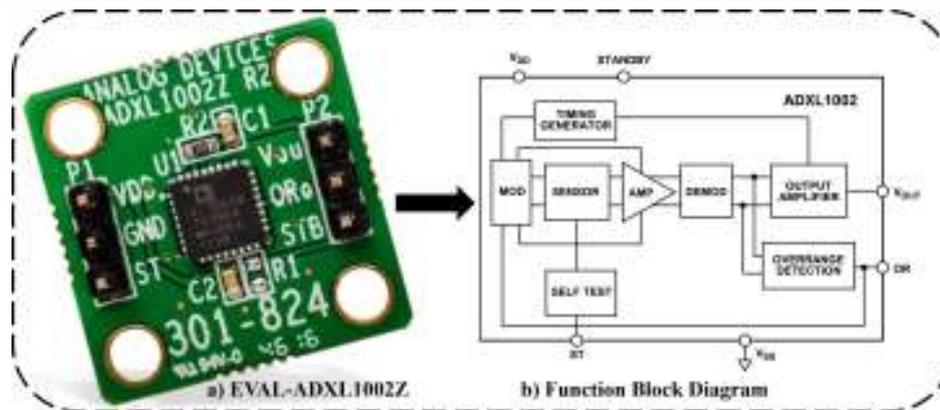


Figure 2.1: EVAL-ADXL1002Z accelerometer and its function block diagram.

withstands dynamic environments with significant vibration, such as HVAC systems and heavy machinery. To achieve peak performance, system noise, mounting, and signal conditioning must be carefully considered. Supply voltage noise impacts system noise because the ADXL1002's analog output is ratiometric, meaning supply voltage fluctuations affect the output. A properly decoupled, stable supply voltage is crucial for powering the ADXL1002 and providing a reference voltage for digitization. Overrange stimuli affect the output signal, and an overload indicator feature helps manage critical conditions in intelligent measurement systems.

The ADXL1002 operates in two modes: measurement mode and standby mode. Measurement mode is the default operational state, where the accelerometer continuously monitors acceleration along its sensitive axis, typically consuming 1.0 mA with a 5.0 V supply. Standby mode suspends measurement activities and significantly reduces current consumption to 225 μA , typical for a 5.0 V supply, with a transition time of less than 50 μs to switch back to measurement mode [2].

Appendix A provides a comprehensive overview of the ADXL1002 accelerometer, detailing its optimal performance characteristics, operational principles, output signal bandwidth, and key features. This appendix serves as a reference for understanding the

Table 2.1: ADXL1002 pin function description.

Pin No.	Mnemonic	Description
1 to 9, 31, 32	NIC	Not Internally Connected.
10, 11, 17 to 19, 21 to 26, 29	DNC	Do No Connect. Leave unconnected
12	V_{DD}	3.3 V to 5.25 V Supply Voltage.
13, 14, 27, 28	V_{SS}	Supply Ground.
15	STANDBY	Standby mode Input, Active High.
16	ST	Self Test Input, Active High.
20	OR	Overrange Output.
30	V_{OUT}	Analog Output Voltage.

accelerometer’s capabilities and its role in signal acquisition for the research. Moreover, Table 2.1 provides a comprehensive description of each pin function, essential for proper integration into various applications.

2.2.2 BeagleBone Black

The BeagleBone Black is a versatile, cost-effective, open-source development platform widely adopted by developers and supported by a robust community [89]. Its flexibility and extensive functionality make it suitable for a wide range of applications, from simple projects to complex industrial systems. The platform is particularly well-suited for sensor interfacing in applications such as vibration measurement systems. Central to the BeagleBone Black is its ARM Cortex-A8 processor, which operates at 1 GHz. This processor, combined with 512 MB of DDR3 RAM and 4 GB of on-board eMMC storage, provides ample computational power and storage capacity for diverse projects (Figure 2.2) [89]. The BeagleBone Black operates on a Linux-based operating system, typically the Debian distribution. This OS provides a robust and versatile environment for software development. A wealth of libraries and development tools are available, which simplifies coding and debugging processes. Furthermore, the board supports a wide array of programming languages, including Python, C, and JavaScript, accommodating developers with varying levels of expertise [89].

The BeagleBone Black incorporates several key components that collectively contribute to its functionality and performance. At its core, the board is powered by the Sitara AM3358BZCZ100 processor, which operates as the central processing unit (CPU). This processor, based on the ARM Cortex-A8 architecture and running at 1

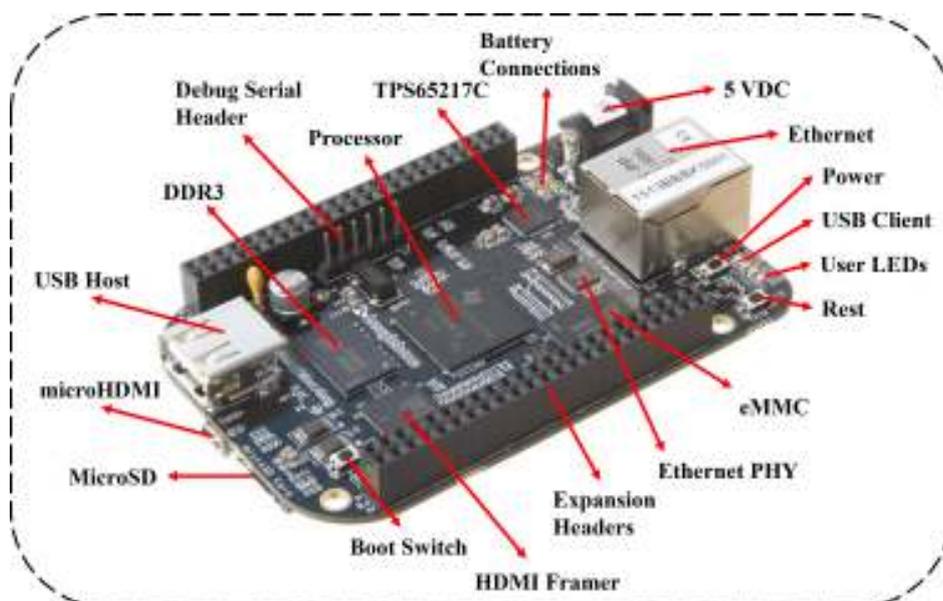


Figure 2.2: BeagleBone Black with its key components.

GHz, forms the computational backbone of the platform, capable of executing a variety of tasks ranging from basic input/output operations to complex computing tasks. The BeagleBone Black is equipped with either Micron 512MB DDR3L or Kingston 512MB DDR3 RAM modules. These dual data rate memory components ensure efficient data handling and processing, crucial for multitasking and handling concurrent operations within applications running on the board [89].

Appendix B details the key features, specifications, and essential components of the BeagleBone Black, including the power module, I/Os, LEDs, and networking. Figure 2.2 visually represents the layout and interconnections of these components, illustrating their roles in enhancing the functionality and versatility of this development platform.

2.2.3 Interfacing Results

Figure 2.3 presents the block diagram of the mobile device, illustrating the integration of ADXL1002 accelerometers with a BeagleBone Black controller and a power module. The power module provides the appropriate voltage levels to all system components, processes the voltage signals from the accelerometers, and operates a bistable button with an indicator lamp. The power module is designed with protective elements against excessive or negative input voltage, which could potentially damage other system components. The power supply output is stabilized at 5 V and further filtered to eliminate interference. Figure 2.3 provides a comprehensive block diagram of the described system architecture.

At the core of the system is the BeagleBone Black single-board computer, which serves as the central computing unit. The built-in analog-to-digital converter (ADC) in the BeagleBone Black samples the data provided by the ADXL1002 accelerometers. These accelerometers are connected to the 3.3V and GND pins of the computer using

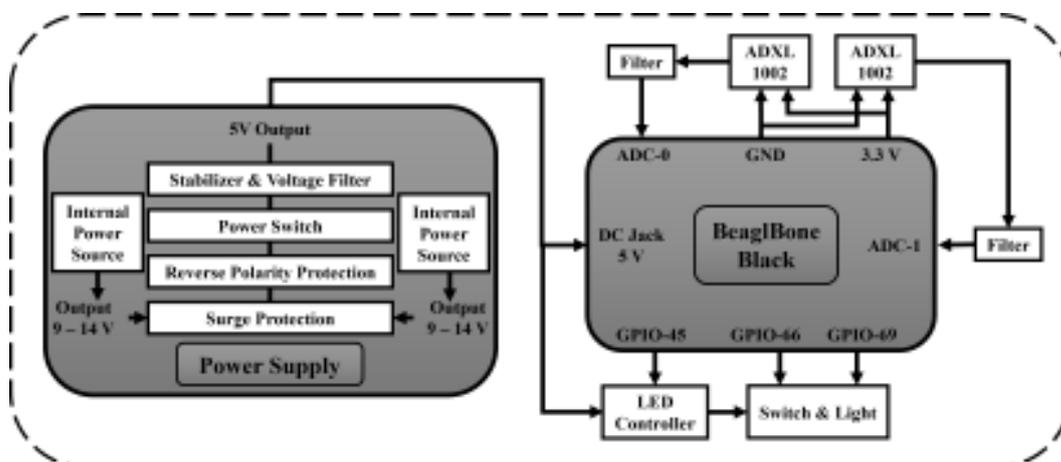


Figure 2.3: Block diagram representing interface of ADXL1002 accelerometer with BeagleBone Black and its power module.

a three-wire shielded cable, minimizing the impact of external electromagnetic interference. The data from the accelerometers is transmitted via the remaining wire to a second-order analog anti-aliasing filter, as depicted in Figure 2.3.

The voltage supply module is designed to produce a stable 5V output, ensuring the uninterrupted operation of the vibration measurement system. To achieve this, an LM7805 voltage regulator is employed to convert an input voltage of 12V into a regulated 5V output. The power supply module is implemented on a printed circuit board (PCB), as illustrated in Figure (2.4). The system utilizes two power sources, labeled V1 and V2, where V1 represents the internal voltage source and V2 denotes the external voltage source. Diodes D1, D2, D3, D4, and D5 are included in the design to protect against overvoltage and reverse voltage conditions.

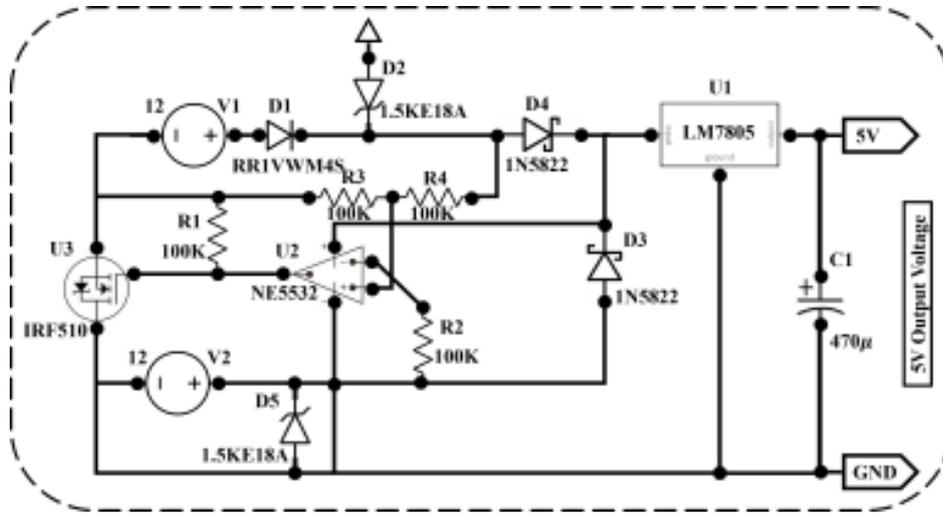


Figure 2.4: Power supply module.

An operational amplifier (U2) is utilized to compare the voltages from the internal and external sources. If the external voltage source exceeds half of the internal voltage source, the output state of U2 transitions to high. When the output of U2 is high, the N-channel MOSFET (U3) is activated, allowing the system to utilize the external voltage source. Conversely, when U3 is turned off, the internal power source is employed. In both scenarios, the LM7805 regulator (U1) continuously converts the 12V input to a stable 5V output. Additionally, the output capacitor (C1) plays a crucial role in stabilizing the power supply, ensuring consistent performance of the vibration measurement system.

Figure (2.5) illustrates the filter and voltage divider configuration. The primary function of the filter is to provide enhanced anti-aliasing filtration, especially considering disturbances generated between the accelerometer and the ADC. According to the ADXL1002 module documentation [2], the module includes a built-in first-order analog filter. However, it is advisable to implement an additional low-pass filter for enhanced performance. The built-in RC filter, indicated by a dashed line in Figure (2.5),

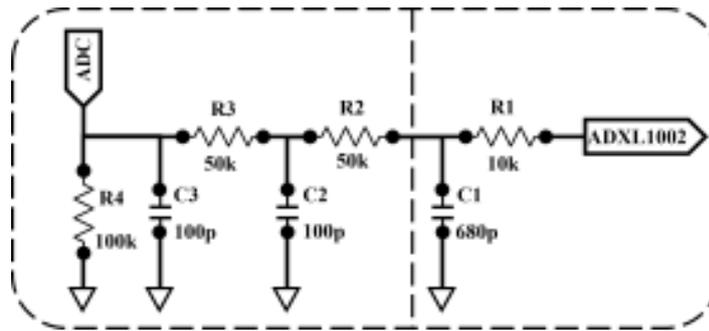


Figure 2.5: Filter with voltage divider.

is situated at a significant distance from the analog-to-digital converter (ADC). This placement is suboptimal as it allows interference generated in the connecting cables to affect the sampled signal.

The designed filter must exhibit a linear characteristic within the passband and a steep roll-off outside it. Although a high-order filter would meet these requirements most effectively, it would also increase the layout's cost and complicate placement on the PCB. To address this, the signal is oversampled with a low-order filter, followed by digital processing at a later stage. In this study, an additional second-order RC filter was employed in conjunction with the existing filter to meet the necessary requirements.

This filter is connected to a voltage divider to lower the voltage level before sampling. This step is essential because the ADC range is limited to 1.8 V, whereas the ADXL1002 output can reach approximately 3.3 V with the applied power supply. The ADC inputs on the BeagleBone Black, labeled ADC 0 and 1, are directly connected to the output of the voltage divider, ensuring the signal is within the acceptable range for accurate sampling.

Additionally, the system includes a button diode controlled by a transistor, necessitated by the low current capacity of the processor's GPIO outputs, which range from 4 mA to 6 mA depending on the pin number, according to the documentation [89]. This current level is insufficient for the diode to accurately indicate the device's operating state. The diode control system is connected to the GPIO 45 output of the BeagleBone

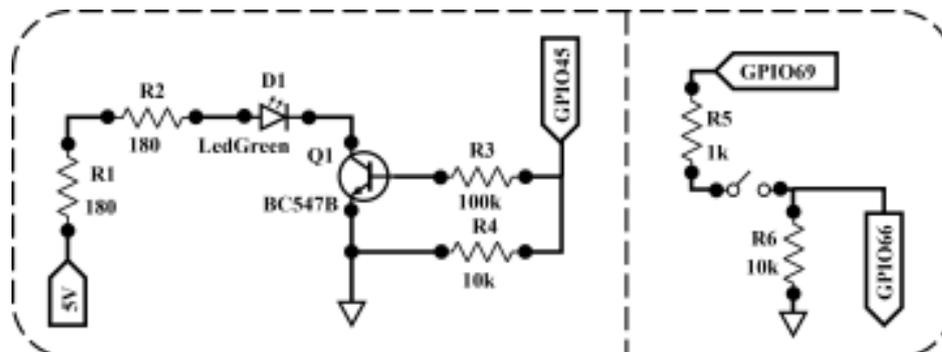


Figure 2.6: LED and switch module.

Black, which controls it, and to the button itself. When pressed, the button shorts the circuit between the GPIO 69 output and the GPIO 66 input as shown in Figure 2.6.

2.3 Calibration

This section discusses the calibration process of the low-cost measurement system based on ADXL1002 MEMS accelerometers, as detailed in [90]. The calibration involved mounting the accelerometers on a vibration exciter. Using a function generator, various frequencies and acceleration signals were applied to the system. The recorded data was analyzed to compare the reference and measured frequencies, evaluate sensitivity across different frequencies, and assess the impact of acceleration on sensor sensitivity. These steps ensured the accuracy and reliability of the ADXL1002 MEMS accelerometers for high-sensitivity vibration measurement applications.

The calibration results are published in the following research paper *Calibration of a High Sampling Frequency MEMS-Based Vibration Measurement System*, presented and published in the conference named *XXI Polish Control Conference*.

2.3.1 Laboratory Setup

The Vibration Laboratory at the Department of Measurements and Control Systems of the Silesian University of Technology in Gliwice was utilized for the calibration experiments. The experimental setup, depicted in Figure 2.7, includes several critical components: a function generator, power amplifier, vibration exciter, measuring amplifier, MEMS accelerometers, and the BeagleBone Black. A detailed description of each piece of hardware used in this experiment is provided in Table 2.2.

Table 2.2: List of Hardware and Description.

Hardware	Type	Description
Function Generator	SFG-2100	Generate vibration signals with different frequencies and amplitudes
Power Amplifier	Type-2706	Enhance the power of the generated vibration signals by function generator
Vibration Exciter	Type-4809	Produce vibrations at specific frequency and amplitude given by the input vibration signal
Measuring Amplifier	Type-2525	Display measurement data as well as selecting setup and measurement parameters
MEMS Accelerometer	ADXL1002	Sense the vibration signals from the vibration exciter and send an output to the microcontroller
BeagleBone Black	–	Record the input vibration signal and save it to the SSD card for further processing

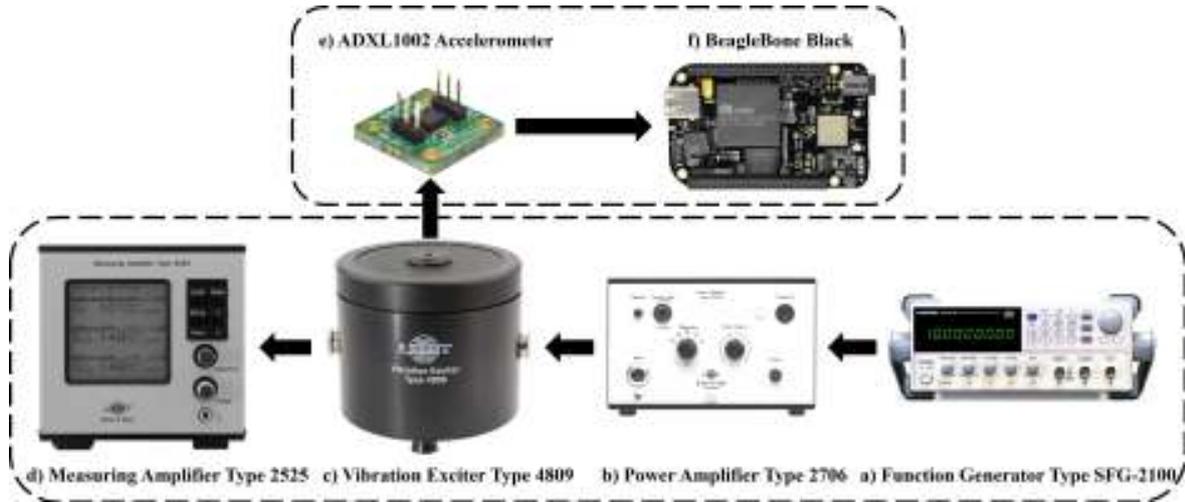


Figure 2.7: Laboratory set for calibration of ADXL1002 accelerometer.

The first component of the experimental setup is the function generator, which is responsible for producing sinusoidal signals of varying frequencies and amplitudes. These signals serve as the input to the system and are essential for simulating different vibration scenarios. The generated signals are then fed into the power amplifier, which boosts their power to levels suitable for driving the vibration exciter.

The vibration exciter, driven by the amplified signals, generates mechanical vibrations at specific frequencies and amplitudes. This piece of equipment is crucial for creating a controlled vibration environment necessary for the calibration of the accelerometers. The vibrations produced by the exciter are measured by two ADXL1002 MEMS accelerometers, which are mounted directly on the vibration exciter. These accelerometers are tasked with capturing the vibration signals and converting them into electrical signals that can be processed further.

The measuring amplifier is used to display the measurement data in real-time and allows for the selection and adjustment of various setup and measurement parameters. This enables precise control and monitoring of the experimental conditions, ensuring that the calibration process is accurate and reliable.

The electrical signals from the ADXL1002 accelerometers are interfaced with the BeagleBone Black, a single-board computer known for its flexibility and computational power. The BeagleBone Black is equipped with analog-to-digital converters (ADCs) that sample the incoming signals from the accelerometers. These digital signals are then recorded and stored on an SSD card for further analysis.

2.3.2 Vibration Dataset

In the course of the experiment, two distinct datasets were meticulously recorded. The first dataset captures the system's response at a constant acceleration of 1m/s^2 across 30 different frequencies. This dataset is crucial for analyzing the behavior of the

accelerometer under a uniform acceleration condition while varying the frequency. Figure 2.8 illustrates the recorded data, highlighting the relationship between the different reference frequencies and the constant acceleration of 1 m/s^2 . The figure specifically shows the frequencies at which the vibration signals were recorded, providing a clear visualization of the accelerometer’s performance at these set points.

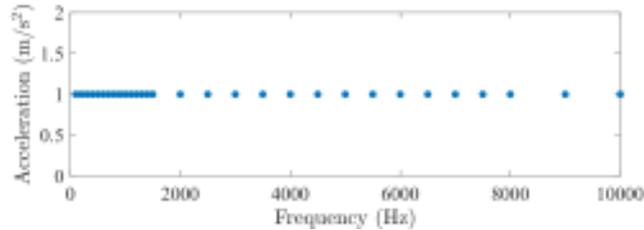


Figure 2.8: Dataset recorded with ADXL1002 accelerometer at 1 m/s^2 acceleration.

The second dataset was compiled by recording the system’s response at specific frequencies—500 Hz, 1kHz, 2kHz, and 5kHz—while varying the acceleration levels. This dataset provides insights into how the accelerometer performs under different accelerations at set frequencies. Figure 2.9 presents the data, showcasing the accelerometer’s output across these frequencies and varying acceleration levels.

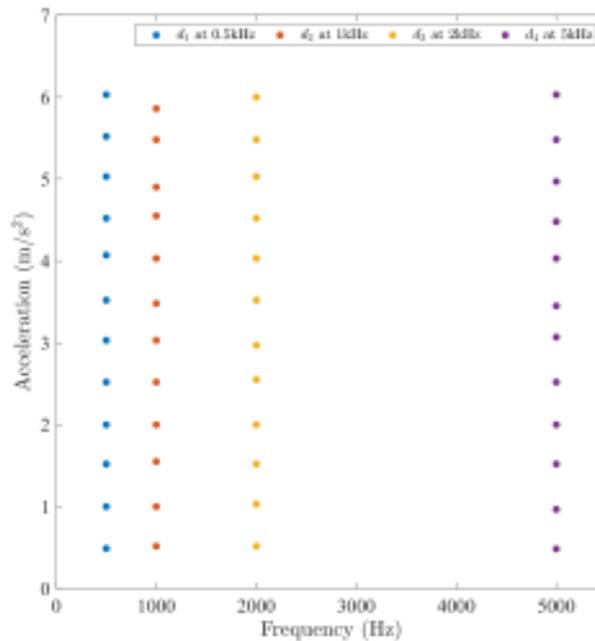


Figure 2.9: Dataset recorded with ADXL1002 accelerometer at various acceleration.

The experiment was methodically repeated multiple times to ensure the reliability and accuracy of the data. The recorded datasets offer a comprehensive view of the accelerometer’s performance, enabling a thorough calibration and validation pro-

cess. The detailed analysis of these datasets is essential for understanding the sensor's sensitivity and accuracy under different operating conditions.

Figure 2.10 shows the time-domain vibration signals recorded under various frequencies: 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 1100, 1200, 1300, 1400, 1500, 2000, 2500, 3000, 3500, 4000, 4500, 5000, 5500, 6000, 6500, 7000, 7500, 8000, 9000, and 10000 Hz. It is noted that Figure 2.10 represents only a portion of the recorded vibration signals because of better and clearer visualization. Furthermore, it can also be seen that low-frequency signals are more distorted than high-frequency signals.

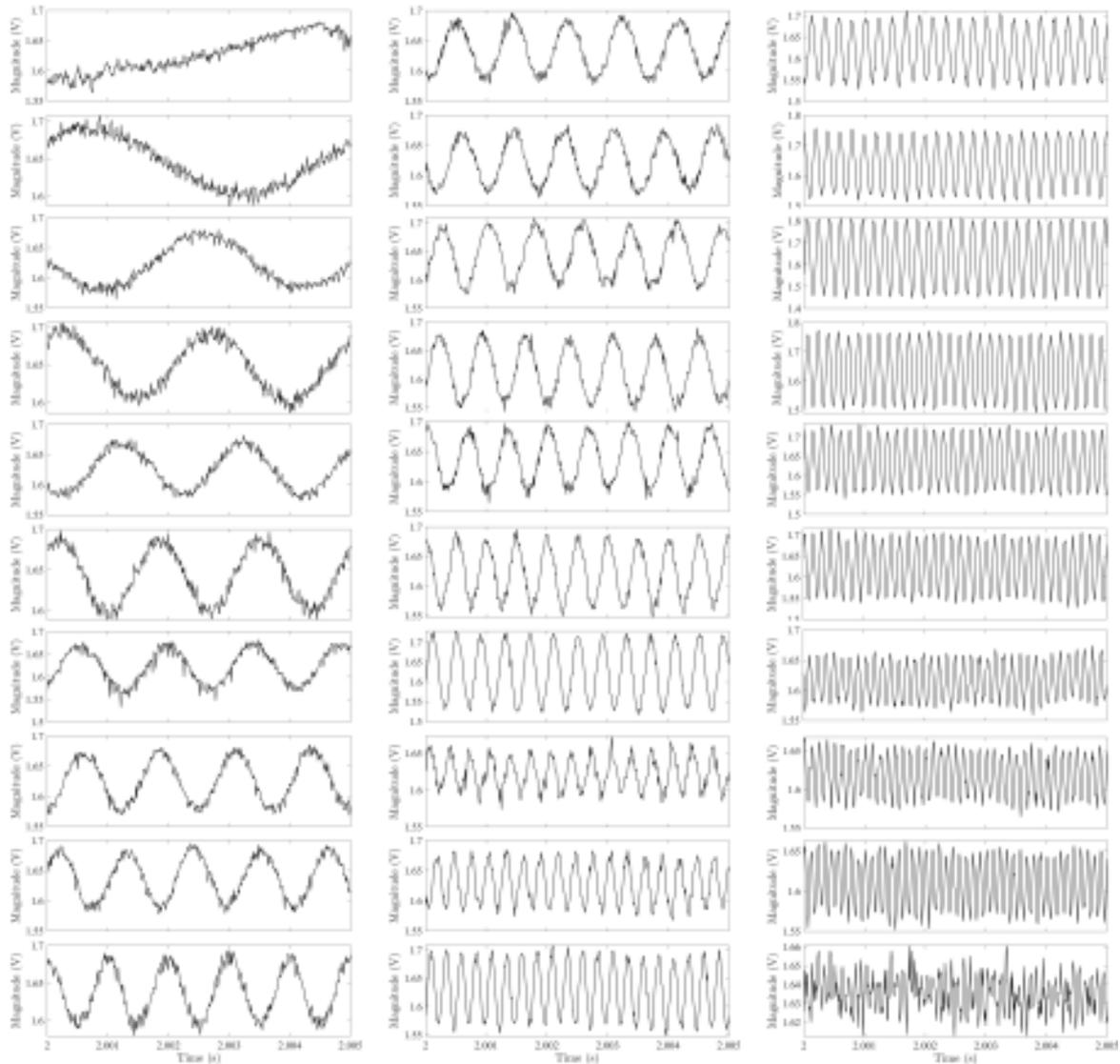


Figure 2.10: Time-domain representation of the dataset recorded with ADXL1002 accelerometer.

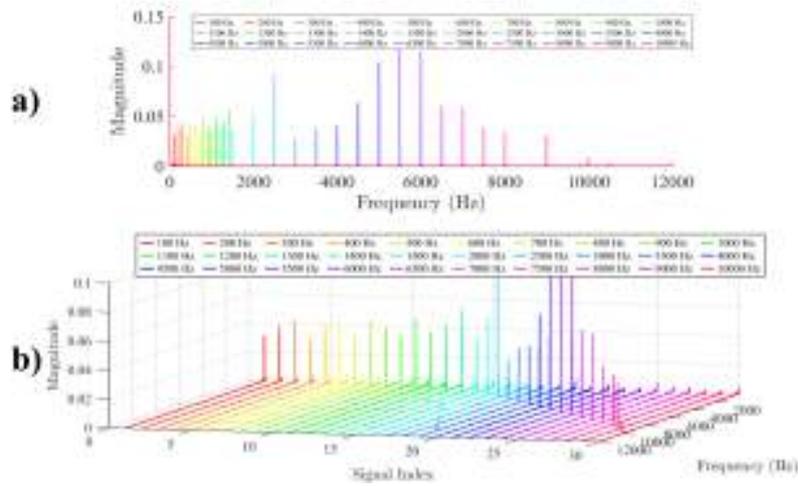


Figure 2.11: Frequency-domain representation of the dataset recorded with ADXL1002 accelerometer.

In addition to the time-domain representation, the frequency-domain representation of the vibration signals was analyzed using Fast Fourier Transform (FFT). Figure 2.11 illustrates the frequency-domain spectra corresponding to the same set of vibration signals. The FFT analysis provides a clear view of the signal components across different frequencies, enabling a more detailed assessment of the accelerometer's response. This representation helps in identifying the dominant frequency components and any potential harmonic distortions present in the signals.

2.3.3 Calibration Results

The acquired datasets described in the previous section were used to illustrate the vibration characteristics of the low-cost ADXL100x accelerometers. Two accelerometers were mounted on the vibration exciter simultaneously to record the vibration datasets. Two datasets were constructed as described above: the first dataset was recorded at different frequencies with constant acceleration as shown in Figure 2.8 whereas the second dataset consists of a vector of four frequencies i.e., 0.5 kHz, 1 kHz, 2 kHz, and 5 kHz and each frequency data was recorded with different levels of acceleration values as shown in figure 2.9. Three types of characteristics were computed from the recorded datasets and listed as follows:

- **Characteristic # 1:** Frequency characteristic includes the relationship between reference frequencies and recorded frequencies.
- **Characteristic # 2:** Sensitivity characteristic at different frequencies consists of a relationship between sensitivity and different frequencies.
- **Characteristic # 3:** Sensitivity characteristic at different accelerations consists of a relationship between sensitivity and different accelerations.

Signal processing methods were applied to the recorded data using MATLAB to

study the vibration characteristics and calibrate the accelerometers. This involved comparing the recorded frequencies with the reference frequencies generated by the function generator. The calibration process aimed to minimize any discrepancies between these frequencies, ensuring the accuracy of the accelerometers. Additionally, the sensitivity characteristics of the ADXL1002 accelerometers were examined across different frequencies to confirm their performance under varying conditions.

The results of this calibration process have been previously published in a conference proceeding [90], demonstrating the effectiveness and reliability of the methods used. This setup and procedure not only ensure precise calibration of the ADXL1002 accelerometers but also validate their suitability for use in vibration measurement systems.

The reference and recorded measured frequencies are plotted together to visualize the accuracy of the ADXL1002 accelerometers in Figure (2.12). These frequencies are measured from the 30 recorded vibration signals shown above in this chapter. The red dots show the reference frequencies whereas the block circles show the recorded frequencies.

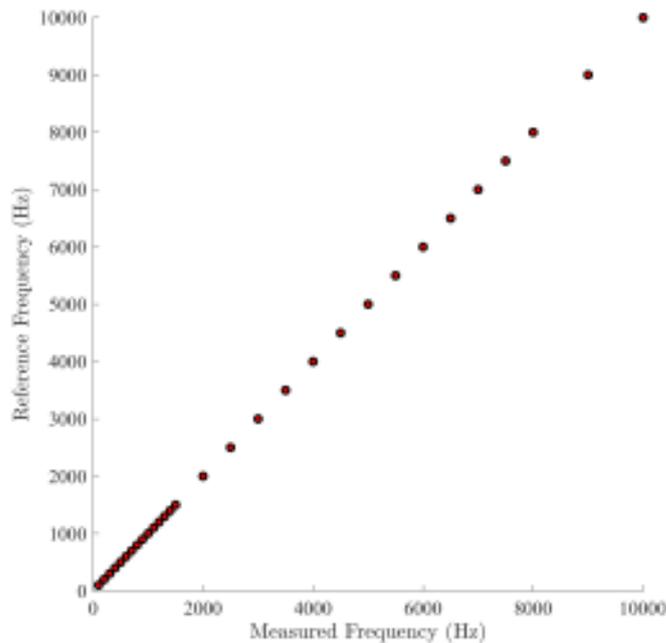


Figure 2.12: Relationship graph between reference frequencies of the vibration shaker and measured frequencies of the ADXL1002 accelerometer.

It is noted that because of minor differences in the reference and measured frequencies, the plot shown in Figure (2.12) neglects the errors among them. However, Figure (2.13) illustrates the errors among references and measured frequencies and from this figure, it can be verified that the error is minor. In this figure, a dotted line representing the upper bound limit and the lower bound limit is plotted at 1.5 and 1.7, which means

the error signal lies within a numerical value of 0.2. This shows the performance of the low-cost ADXL100x accelerometer. The minor errors indicate the efficiency of the ADXL1002 accelerometers in recording the vibration signals with specific frequencies efficiently.

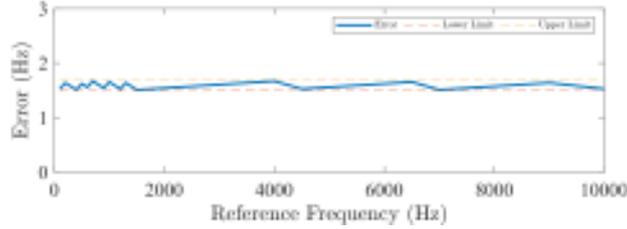


Figure 2.13: Errors among reference frequencies of the vibration shaker and measured frequencies of the ADXL1002 accelerometer.

Furthermore, to examine the second and third characteristics, sensitivity was computed using the following equation:

$$S_i = \frac{\text{RMS}(x(f_i))}{g_i} \quad (2.1)$$

where $x(f_i)$ is the recorded time-domain vibration signal at a specific frequency f_i with $i = 1, 2, 3, \dots, n$, g_i is the acceleration of the vibration signals, $\text{RMS}(\cdot)$ is the root-mean-square of the signal, and S_i is the sensitivity index. To plot the sensitivity at different frequencies, the first dataset was used. Figure (2.14) shows the sensitivities of both accelerometers at different frequencies. Furthermore, it can be seen that the sensitivities of both accelerometers are almost similar that shows both accelerometers perform well with similar characteristics.

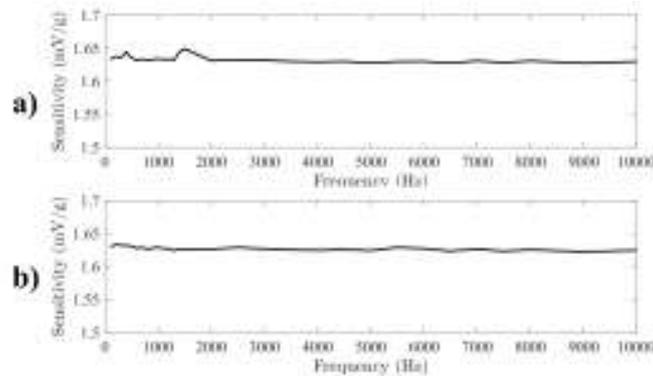


Figure 2.14: Sensitivities of both accelerometers at different frequencies a) sensitivity of the first accelerometer and b) sensitivity of the second accelerometer.

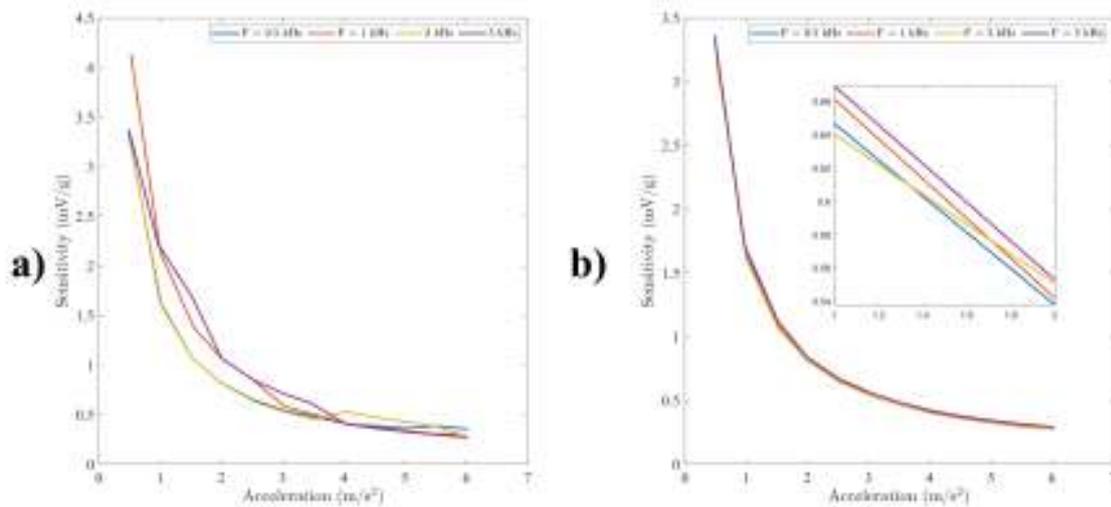


Figure 2.15: Sensitivity versus acceleration at different frequencies: a) characteristic of the first accelerometer, and b) characteristic of the second accelerometer.

The second dataset was recorded keeping the constant frequency and varying the acceleration values. The sensitivity for each signal was computed using the equation (2.1). To illustrate the third characteristic, a plot between sensitivity and acceleration was constructed for both accelerometers as shown in Figure 2.15. The graphs show that the sensitivity is inversely proportional to the acceleration, indicating that higher gravitational levels result in reduced sensitivity. All the results of vibration characteristics show that the low-cost ADXL1002 accelerometers are efficient for sensitive applications and diagnose the frequencies accurately. Utilizing the developed prototype of the ADXL1002 accelerometer interfaced with the BeagleBone Black, the calibration process lays the foundation for advanced applications in misfire detection within vehicle engine vibration data.

2.4 Conclusion

The work presented in this chapter focused on interfacing and calibrating the ADXL1002 accelerometer with the BeagleBone Black single-board computer. The process underscored the significance of a well-integrated system design and robust calibration methods for achieving reliable vibration measurements in industrial and automotive applications.

In the interfacing section, the study detailed the design and integration of essential modules, including the power supply unit, filtering systems, and voltage divider circuits, to ensure stable and precise data acquisition. The use of the ADXL1002 accelerometer, known for its high sensitivity and broad frequency range, proved effective in capturing detailed vibration signals, while the BeagleBone Black provided a versa-

tile and computationally capable platform for real-time data processing and storage. Key design considerations, such as mitigating electromagnetic interference and ensuring optimal analog-to-digital conversion, further enhanced the system's accuracy and performance.

Calibration was conducted in a controlled laboratory environment to validate the accelerometer's precision and reliability across various frequencies and acceleration levels. The systematic approach employed a vibration exciter and function generator to simulate diverse operational scenarios, enabling the assessment of the sensor's sensitivity and frequency response. The resulting datasets revealed consistent performance of the ADXL1002, with minimal errors observed between reference and measured frequencies. This calibration not only verified the accelerometer's capability to deliver accurate measurements but also demonstrated its suitability for applications requiring high sensitivity, such as machinery condition monitoring and vehicle misfire diagnosis.

Moreover, the study highlighted the accelerometer's inverse sensitivity relationship with acceleration levels, emphasizing the need for careful consideration of operational conditions in practical applications. The sensitivity analysis also confirmed the uniformity in performance between multiple accelerometer units, ensuring the system's scalability and reliability.

In conclusion, this chapter establishes a comprehensive methodology for designing, interfacing, and calibrating a low-cost vibration measurement system. The integration of the ADXL1002 accelerometer with the BeagleBone Black has proven effective for capturing precise vibration signals, offering a scalable solution for industrial and automotive diagnostics. The insights gained from this work pave the way for future developments, including the implementation of advanced signal processing techniques and real-time fault diagnosis systems, ultimately contributing to more efficient and reliable condition monitoring technologies.

Chapter 3

Vibration Datasets

3.1 Introduction

This chapter presents a comprehensive overview of the vibration datasets used in the study, specifically focusing on datasets collected from vehicle engine and rotating element bearings. The significance of these datasets lies in their potential to enhance the understanding of machinery health and operational conditions through the analysis of vibration signals. By leveraging advanced sensors and data acquisition techniques, the datasets provide invaluable insights into the dynamic behavior of mechanical systems under various operational scenarios.

The first section details the vibration dataset collected from a vehicle engine using the ADXL1002 accelerometer. The data collection was meticulously conducted under controlled experimental conditions to ensure the accuracy and reliability of the measurements. The experimental setup involved capturing signals from the vehicle engine under different states, including healthy and misfire conditions, various loading scenarios (unloaded, half-loaded, and fully loaded), and a range of engine speeds (RPMs) of 1500 RPM, 2500 RPM, and 3000 RPM. The experiments were carried out at the Vehicle Vibration Laboratory, Department of Transportation, Silesian University of Technology in Katowice.

In addition to the vehicle engine vibration dataset, this chapter explores other significant vibration datasets. Bearing vibration datasets play a crucial role in the field of condition monitoring and fault diagnosis for rotating machinery. These datasets, widely documented in the literature, provide valuable insights into the dynamic behavior of bearings under varying operational conditions, including healthy states and specific fault scenarios. They serve as a foundation for developing and validating advanced diagnostic algorithms aimed at predicting failures and optimizing maintenance strategies. Among the prominent datasets, the Intelligent Maintenance Systems (IMF) Test-to-Failure Vibration Dataset from NASA and the MFPT Fault Dataset are extensively

used for research and industrial applications. These datasets offer a range of scenarios, such as bearing operation under nominal loads, outer and inner race faults, and varying loads, making them indispensable for exploring the relationships between vibration characteristics and mechanical faults. Their accessibility and detailed structure allow researchers to refine and test fault detection models, contributing significantly to advancements in condition-based maintenance systems.

3.2 Vibration Dataset from Vehicle Engine using ADXL1002 Accelerometer

3.2.1 Experimental Setup

The vibration data acquisition was conducted using the ADXL1002 accelerometer, capturing signals from a vehicle engine under various operational conditions. The conditions included healthy and misfire states, different loading scenarios (unloaded, half-loaded, and fully loaded), and a range of engine speeds (RPMs) including 1500 RPM, 2500 RPM, and 3000 RPM. This study aimed to analyze the vibration characteristics associated with these differing conditions.

The experiments were carried out at the Vehicle Vibration Laboratory, within the Department of Transportation at the Silesian University of Technology in Katowice. The laboratory setup, as shown in Figure 3.1, meticulously illustrates the procedural arrangement and the input conditions applied prior to data recording.

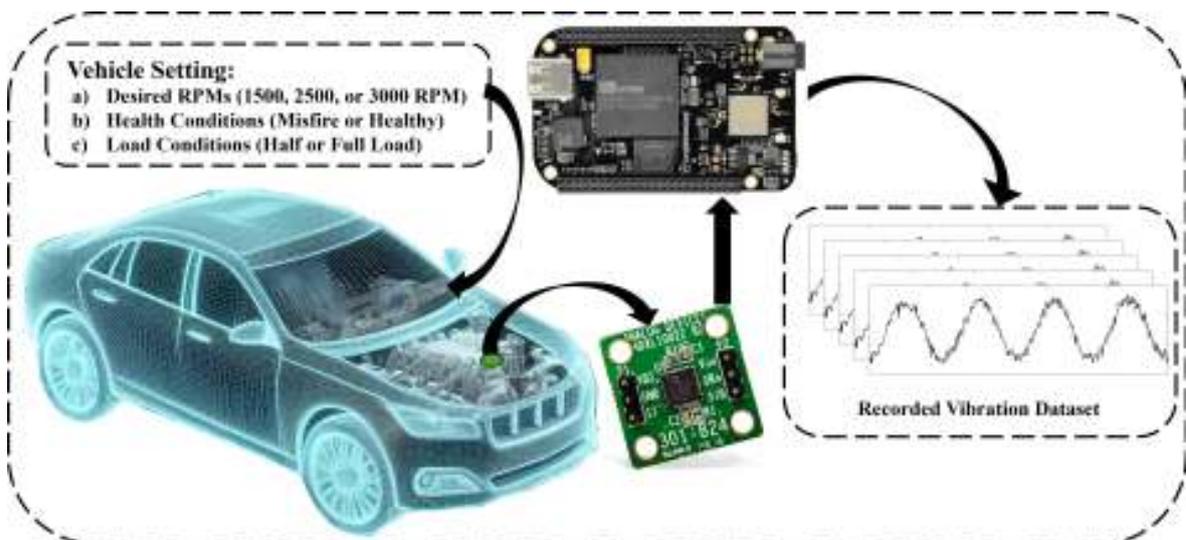


Figure 3.1: Placement of ADXL1002 Accelerometer on Vehicle Engine for Vibration Data Collection.

The experimental conditions were carefully controlled and defined as follows:

- **Engine Speed (RPMs):** The engine was operated at three distinct speeds: 1500 RPM, 2500 RPM, and 3000 RPM.
- **Engine Health Status:** The engine was tested under two health conditions:
 - **Healthy Condition:** The engine operates normally without any faults.
 - **Misfire Condition:** The engine experiences misfiring, simulating a common fault condition.
- **Loading Conditions:** The engine was subjected to three loading scenarios:
 - **Unloaded Condition:** The engine runs without any additional load.
 - **Half Load Condition:** The engine operates with a medium load.
 - **Full Load Condition:** The engine operates under maximum load.

The ADXL1002 accelerometer was securely mounted on the engine using a magnetic contact, ensuring stable and accurate vibration measurement. The accelerometer was interfaced with a BeagleBone Black microcontroller for data acquisition, as detailed in the preceding chapter. Figure 3.2 depicts the precise placement of the accelerometer on the vehicle engine, highlighting its strategic positioning for optimal data capture.

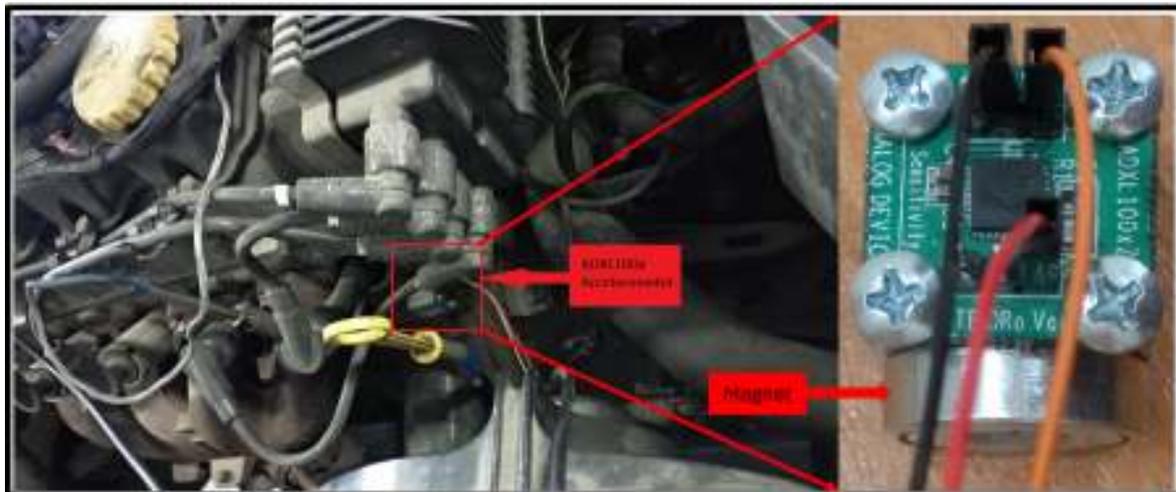


Figure 3.2: Placement of ADXL1002 Accelerometer on Vehicle Engine for Vibration Data Collection.

The collected vibration data were stored on an SSD card, ensuring reliable and efficient data handling. This storage method facilitated the subsequent analysis of the vibration signals under the various test conditions.

The detailed setup and controlled conditions for the vibration data collection experiment are crucial for ensuring the accuracy and reliability of the data. The experimental design allows for a comprehensive analysis of the impact of different operational conditions on vehicle engine vibrations, contributing valuable insights for fault detection and engine performance optimization.

3.2.2 Vehicle Engine Dataset

The vehicle engine vibration dataset given in Table 3.1 offers a structured overview of engine conditions across various RPMs, loads, and misfire occurrences. Each entry in the table corresponds to a specific engine scenario, denoted by RPM (Revolutions Per Minute), load intensity, and misfire status. The dataset captures instances across different RPMs, including 1500, 2500, and 3000, coupled with load conditions ranging from no load to half load and full load. Moreover, the dataset signifies whether a misfire was present or absent in each specific scenario. This dataset serves as a comprehensive representation of engine performance variations under different operational settings, allowing for in-depth analysis and exploration of how RPM, load, and misfire interrelate within the context of vehicle vibrations.

Table 3.1: Vehicle engine vibration dataset.

RMS	Frequency	Load	Misfire
1500 RPM	25 Hz	no load	no misfire
3000 RPM	50 Hz	no load	no misfire
1500 RPM	25 Hz	half load	no misfire
1500 RPM	25 Hz	full load	no misfire
2500 RPM	41.667 Hz	half load	no misfire
2500 RPM	41.667 Hz	full load	no misfire
3000 RPM	50 Hz	half load	no misfire
3000 RPM	50 Hz	full load	no misfire
1500 RPM	25 Hz	half load	misfire
1500 RPM	25 Hz	full load	misfire
2500 RPM	41.667 Hz	half load	misfire
2500 RPM	41.667 Hz	full load	misfire
3000 RPM	50 Hz	half load	misfire
3000 RPM	50 Hz	full load	misfire

To further elucidate the engine vibration dataset captured through the ADXL1002 accelerometer, a representation in the time domain is crucial for visual comprehension. The vehicle vibration dataset is plotted in a 3D time-domain signal representation to provide a comprehensive visualization of the recorded data in Figure 3.3. This visualization includes 15 distinct vibration signals, corresponding to the various conditions outlined in Table 3.1. The 3D representation effectively captures the complex dynamics of the vehicle engine vibrations under different operational states, allowing for a more intuitive understanding of the variations in the signals.

The selected vibration signals, depicted in Figure 3.4 showcase the temporal characteristics of engine vibrations across distinct operational states. These signals, plotted against time, offer a direct insight into the fluctuations and patterns within the recorded vibrations during various engine conditions, such as RPMs, load levels, and

misfire occurrences.

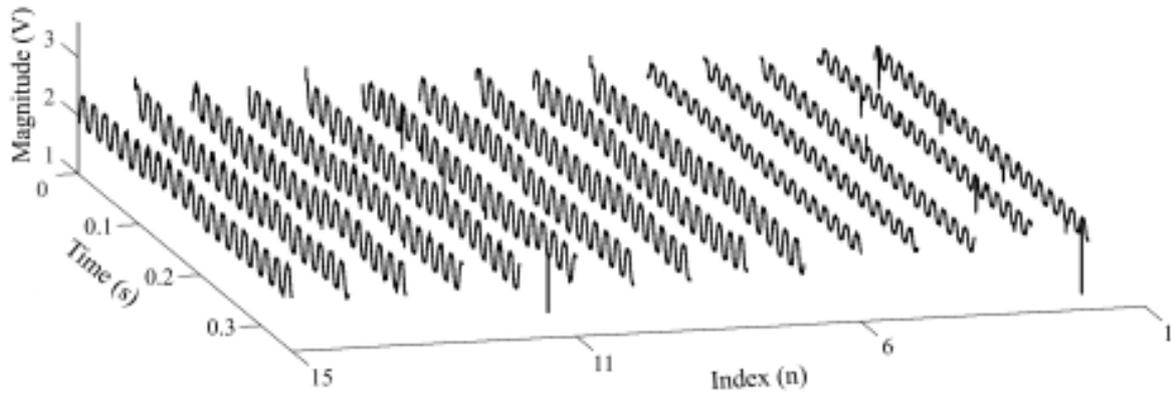


Figure 3.3: 3D representation of the vehicle vibration dataset recorded with ADXL1002 accelerometer.

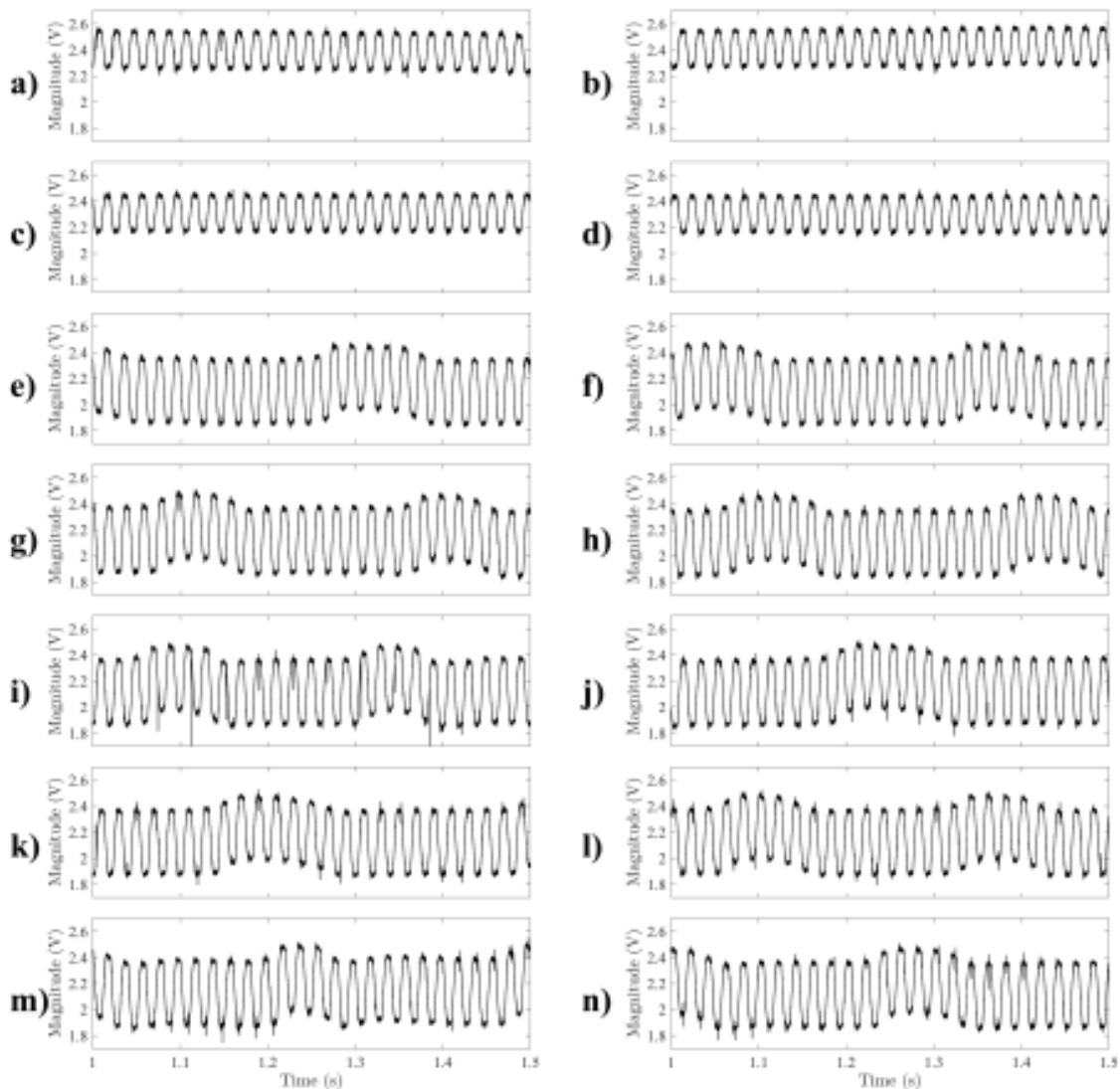


Figure 3.4: Time-domain representation of the vehicle vibration dataset recorded with ADXL1002 accelerometer.

3.3 Bearing Vibration Datasets in the Literature

Bearing vibration datasets provide a rich source of information for the study of fault diagnosis and predictive maintenance techniques. These datasets capture vibration signals from bearings operating under controlled laboratory conditions and real-world scenarios. They are pivotal in understanding how faults develop and propagate over time, enabling the design of robust algorithms for machinery health monitoring. In this study, we focus on two significant datasets: the Test-to-Failure Vibration Dataset and the MFPT Bearing Fault Dataset, both of which have been widely used for research in condition monitoring and diagnostics.

3.3.1 IMF Test-to-Failure Vibration Dataset

The test-to-failure vibration dataset of bearings is a collection of data that records the vibration behavior of bearings under operational conditions until they fail. This dataset is often used in predictive maintenance and condition monitoring research to develop and validate algorithms for fault detection and remaining useful life (RUL) estimation [93, 1].

The IMF bearing vibration dataset comprises data from an experimental setup where four bearings were installed on a shaft. The shaft maintained a constant rotational speed of 2000 RPM, driven by an AC motor coupled via rub belts. A radial load of 6000 lbs was applied to the shaft and bearings using a spring mechanism, and all bearings were force-lubricated. The bearings used were Rexnord ZA-2115 double-row bearings. High Sensitivity Quartz ICP accelerometers (PCB 353B33) were mounted on the bearing housings, with two accelerometers (measuring x- and y-axes) for each bearing in dataset 1, and one accelerometer per bearing for datasets 2 and 3. Failures were observed after the bearings exceeded their designed lifetime of over 100 million revolutions. Figure 3.5 illustrates the experimental set for IMF test-to-failure vibration dataset recording.

The dataset is structured into three sets, each representing a test-to-failure experiment. Each dataset contains files with 1-second snapshots of vibration signals recorded at specific intervals, consisting of 20,480 data points sampled at 20 kHz, facilitated by the NI DAQ Card 6062E. The file names indicate the time of data collection, with larger intervals marking the resumption of experiments on subsequent working days.

Set 1 was recorded from October 22, 2003, to November 25, 2003, encompassing 2,156 files with 8 channels. The channels were arranged as Bearing 1 – Ch 1&2, Bearing 2 – Ch 3&4, Bearing 3 – Ch 5&6, and Bearing 4 – Ch 7&8. Files were recorded every 10 minutes, except for the first 43 files, which were recorded every 5 minutes. This experiment concluded with an inner race defect in bearing 3 and a roller element defect in bearing 4. Set 2, recorded from February 12, 2004, to February 19, 2004, includes

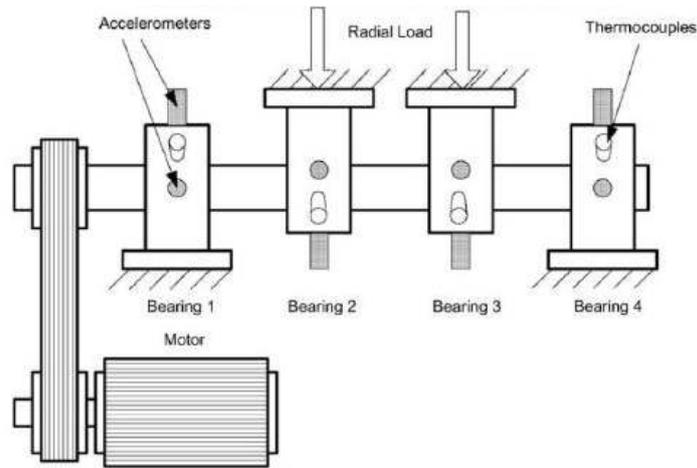


Figure 3.5: Experimental setup for IMF Test-to-Failure Vibration Dataset Recording [1].

984 files with 4 channels arranged as Bearing 1 – Ch 1, Bearing 2 – Ch 2, Bearing 3 – Ch 3, and Bearing 4 – Ch 4, with files recorded every 10 minutes. This set concluded with an outer race failure in bearing 1. Set 3, recorded from March 4, 2004, to April 4, 2004, contains 4,448 files with 4 channels arranged similarly to Set 2. Files were recorded every 10 minutes, and this experiment concluded with an outer race failure in bearing 3. The block diagram of the dataset is illustrated in Figure 3.6.

The dataset is publicly available on Kaggle [94] and the official NASA website, provided in text format and compressed in RAR and ZIP formats. It includes comprehensive documentation (README) by the authors. This dataset was provided by the Center for IMS at the University of Cincinnati. Acknowledgments go to J. Lee, H. Qiu, G. Yu, J. Lin, and Rexnord Technical Services (2007) for the "Bearing Data Set", available at the NASA's Open Data Portal .

3.3.2 MFPT Bearing Fault Dataset

The Condition Based Maintenance (CBM) paradigm aims to enhance reliability and reduce maintenance costs by shifting from scheduled maintenance to condition-based maintenance. This approach minimizes unscheduled maintenance events and, once validated, supports on-condition maintenance. However, CBM technology is still maturing, with significant scope for improvement in algorithms and techniques for reporting and displaying component conditions. Additionally, there are few datasets with known faults available to validate the performance of new algorithms.

To address these challenges, the MFPT Fault Database provides various datasets of known good and faulty conditions for bearings and gears. This dataset is freely available, including example processing code, to encourage researchers and CBM practitioners to advance techniques and accelerate the maturity of CBM systems [95].

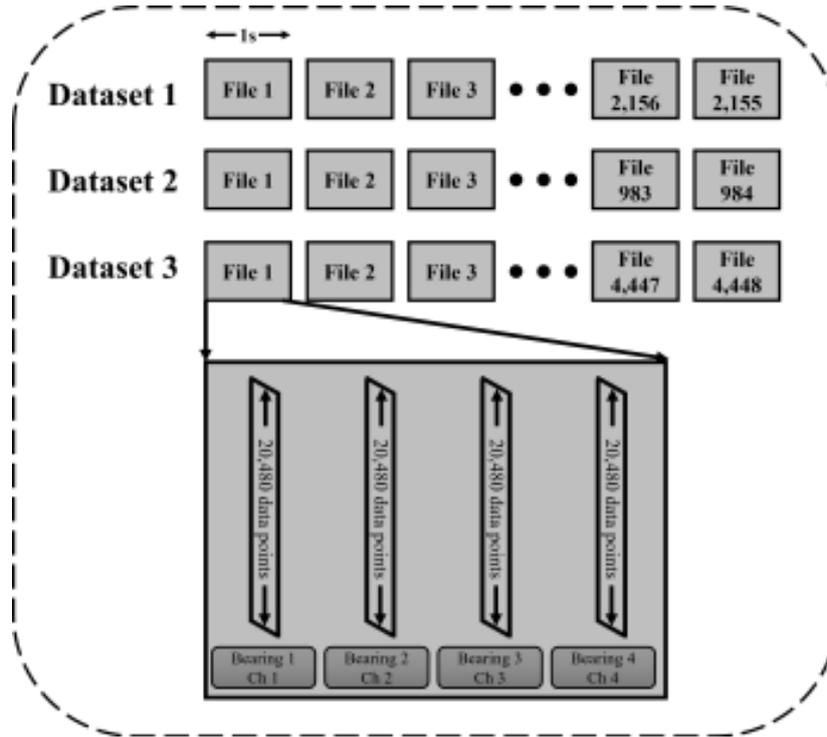


Figure 3.6: IMF Test-to-Failure Dataset Visualization.

The bearing fault dataset supports research into bearing analysis, comprising data from a test rig with a NICE bearing under various conditions, such as nominal bearing data, outer race faults under varying loads, and inner race faults under varying loads. The test rig's bearing has specific parameters: roller diameter, pitch diameter, number of elements, and contact angle. The dataset includes:

1. **Baseline conditions:** 270 lbs load, 25 Hz shaft rate, 97,656 sps sample rate for 6 seconds.
2. **Outer race fault conditions:** 270 lbs load, 25 Hz shaft rate, 97,656 sps sample rate for 6 seconds.
3. **Outer race fault conditions:** under different loads (25 to 300 lbs), 25 Hz shaft rate, 48,828 sps sample rate for 3 seconds.
4. **Inner race fault conditions:** under different loads (0 to 300 lbs), 25 Hz shaft rate, 48,828 sps sample rate for 3 seconds.

Additionally, the dataset includes three real-world fault examples from wind turbine bearings and an oil pump shaft bearing. The data is stored in MATLAB® double-precision binary format, containing load, shaft rate, sample rate, and vibration data.

These datasets are invaluable for researchers and practitioners in the CBM field, facilitating advancements in fault detection and maintenance optimization techniques. By providing access to this data, the MFPT Fault Database aims to foster the development and validation of new algorithms and methodologies, contributing to the maturation and wider adoption of CBM technologies.

3.4 Conclusion

Chapter 3 provides a detailed exploration of vibration datasets, underscoring their critical role in the analysis and diagnosis of mechanical systems. The datasets discussed—ranging from those collected from vehicle engines using advanced accelerometers to widely recognized bearing datasets—highlight the comprehensive approach to studying vibration characteristics under varying operational conditions.

The vehicle engine dataset, meticulously gathered using the ADXL1002 accelerometer, demonstrated the significant influence of engine speed, loading conditions, and misfire states on vibration patterns. By maintaining controlled experimental setups, the dataset offers reliable insights into engine behavior, facilitating the identification of distinct vibration signatures associated with mechanical anomalies. Such datasets not only enhance the understanding of engine dynamics but also serve as foundational resources for developing fault detection strategies.

Similarly, the bearing vibration datasets presented in this chapter, particularly the NASA Test-to-Failure and MFPT Fault datasets, underscore the extensive research and diagnostic potential these resources provide. With detailed scenarios involving healthy and faulty states, varying loads, and operational conditions, these datasets are invaluable for validating diagnostic algorithms and advancing condition-based maintenance techniques.

The discussions in this chapter emphasize the importance of accurate data collection and structured datasets in fostering advancements in predictive maintenance and fault diagnosis methodologies. By integrating experimental rigor with practical relevance, the datasets highlighted here contribute to the broader field of mechanical diagnostics, enabling more effective and efficient maintenance strategies for various applications.

Chapter 4

Digital Signal Processing Methods for Vehicle Diagnostics

4.1 Introduction

The detection and diagnosis of faults in vehicles are crucial for ensuring reliability, performance, and safety. Among various engine faults, misfire and bearing failures are two critical malfunctions that demand prompt attention due to their significant impact on engine operation. Misfire, which occurs when one or more cylinders in an engine fail to ignite, leads to uneven engine performance, increased fuel consumption, higher emissions, and, if unresolved, potential engine damage [96, 97, 12, 98, 99]. Similarly, rotating bearings, which are integral components in various vehicle subsystems, are prone to faults caused by wear, misalignment, or inadequate lubrication. Bearing failures can result in excessive vibrations, noise, and mechanical degradation, ultimately compromising the efficiency and safety of the vehicle. Accurate and timely diagnosis of both misfire and bearing faults is essential to minimize operational disruptions, prevent further damage, and extend the lifespan of engine components.

Digital Signal Processing (DSP) methods have become indispensable in vehicle diagnostics, enabling precise fault detection by analyzing the complex signals generated by various engine components. Vibration signals, in particular, serve as a rich source of diagnostic information, capturing the mechanical responses produced by internal combustion processes and the operation of rotating elements such as bearings. These signals provide critical insights into engine health, making it possible to accurately detect faults such as misfires and bearing failures [100, 101]. Misfires and bearing faults, both manifest distinct signatures within vibration data. DSP methods excel in extracting meaningful features from these signals, enabling the identification of subtle irregularities that signify early-stage misfire or bearing degradation. By leveraging these techniques, engineers can enhance diagnostic accuracy, optimize maintenance

schedules, and improve overall vehicle reliability and safety.

This chapter explores the application of DSP methods for fault diagnosis in vehicle engines, emphasizing the detection of misfires and bearing faults through vibration signal analysis. The chapter synthesizes insights from a range of DSP methods employed in this domain, including filter design for effective noise reduction, spectral analysis for identifying fault-related frequency components, and advanced time-frequency analysis to capture transient and nonstationary features. These methods are essential for isolating the distinct diagnostic signatures of misfires and bearing anomalies, addressing challenges posed by noise and the complexity of real-world vibration data.

Additionally, this chapter incorporates findings from three relevant papers that contribute to the advancement of fault diagnosis in vehicle engines, particularly focusing on misfires and bearing faults. The first two papers, *Early-Stage Fault Diagnosis for Rotating Element Bearing Using Improved Harmony Search Algorithm with Different Fitness Functions*, published in *IEEE Transactions on Instrumentation and Measurement*, and *ARL-Wavelet-BPF Optimization using PSO Algorithm for Bearing Fault Diagnosis*, published in *Archives of Control Sciences*, present innovative filter design techniques for noise reduction and signal enhancement, applied to bearing datasets.

The third paper, *Analysis of the Vehicle Engine Misfires using Frequency-Domain Approaches at Various RPMs with ADXL1002 Accelerometer*, published in *Archives of Acoustics*, directly examines vehicle engine vibration data. This paper employs advanced signal processing techniques, including Fast Fourier Transform (FFT), envelope spectrum (ES) analysis, and the extraction of the first Intrinsic Mode Function (IMF) using Empirical Mode Decomposition (EMD), to identify misfire patterns. By applying these methods to real-world engine vibration data, the paper offers practical insights into their diagnostic potential, highlighting their relevance for accurate misfire detection under varying engine conditions, such as different RPMs. Together, these three papers provide a comprehensive foundation for developing robust diagnostic frameworks that address both misfire and bearing faults, leveraging the power of DSP techniques for enhanced reliability and performance in vehicle engine monitoring.

4.2 Overview of Digital Signal Processing Methods

The accurate diagnosis of faults in mechanical systems, especially in vehicle engines, requires advanced analysis of vibration signal data generated by these systems. DSP methods are widely used for this purpose due to their ability to handle complex and noisy data, isolating critical features that indicate potential malfunctions. These techniques play a vital role in fault detection, particularly for engine misfires and bearing faults, both of which significantly impact vehicle performance and safety. This section provides an overview of DSP methods applied to fault diagnosis, focusing on their use

in vibration-based fault detection for engine misfires and bearing failures.

4.2.1 DSP Methods for Bearing Faults

Bearing faults are prevalent in vehicles and can result in substantial mechanical damage if not identified and addressed in a timely manner. Bearings, as critical components in rotating machinery, typically generate periodic and predictable vibration patterns during normal operation. However, when faults occur—such as wear, misalignment, or insufficient lubrication—the vibration characteristics become irregular, introducing new frequency components or altering the existing ones. These deviations can serve as key indicators of bearing issues, but they are often subtle and masked by noise, making accurate fault diagnosis challenging. To effectively identify these faults, DSP methods are employed to extract the relevant features from the vibration signals, improving the diagnostic accuracy.

Various filtering techniques, including low-pass, high-pass, and band-pass filters, are commonly applied to remove irrelevant frequencies and enhance the fault-related components in the signal [70, 102, 103, 104]. These filters are critical for isolating the fault signatures and ensuring that the diagnostic process focuses on the most relevant aspects of the signal. However, static filter designs with fixed parameters may not be optimal for every scenario, as vibration signals can vary based on factors such as load conditions, speed, and environmental influences. This is where metaheuristic algorithms such as the HSA and Particle Swarm Optimization (PSO) come into play. In this chapter, HSA is combined with kurtosis, spectral kurtosis (SK) and Short-Time Fourier Transform (STFT)-based SK as fitness functions, while PSO is employed with the Continuous Wavelet Transform (CWT) to refine filter parameters. These optimized filters significantly improve the signal-to-noise ratio, enhancing diagnostic accuracy.

4.2.2 DSP Methods for Engine Misfire

Misfire detection in vehicle engines requires the analysis of vibration signals generated by combustion irregularities. When misfires occur, the combustion process becomes incomplete or inconsistent, disrupting the normal vibration patterns of the engine. These disruptions manifest as changes in the frequency and amplitude of the vibration signal, which can be detected through various DSP techniques. Time-domain and frequency-domain analyses are both crucial in diagnosing misfires. While time-domain analysis provides insights into signal amplitude and timing, frequency-domain analysis, including FFT, ES analysis, and SK, reveals the frequency shifts and new components introduced by misfires. A combination of these techniques, along with wavelet-based methods and STFT, is commonly employed to improve fault detection accuracy and capture transient events such as misfires. Additionally, DSP techniques

such as the extraction of the first IMF using EMD offer a refined approach to isolating fault frequencies in engine vibration data.

4.2.3 Challenges in DSP Methods for Vehicle Diagnostics

While DSP techniques offer substantial capabilities for fault diagnosis, their application to vehicle misfire and bearing fault detection presents several challenges:

- **Noise and Interference:** Vibration signals from both bearing and misfire datasets are often contaminated by various sources of noise and interference. In the case of bearing faults, external factors such as mechanical vibrations from other components, road noise, and even temperature fluctuations can interfere with the vibration signals, masking subtle fault signatures. For misfire detection, additional sources of noise include engine idling, irregular fuel combustion, and electrical interference from other vehicle systems. These types of noise can introduce irrelevant frequency components or distort the signal, making it difficult to isolate fault-related features. This interference typically arises due to the complex nature of vibration signals, which are influenced by a variety of operational and environmental conditions, as well as the non-ideal behavior of mechanical and combustion processes.
- **Non-stationary Nature of Signals:** Vibration signals are inherently non-stationary, meaning that their statistical properties change over time. These non-stationary characteristics complicate the analysis, as traditional signal processing methods may fail to capture time-varying features effectively.
- **Complexity of Signal Interpretation:** Interpreting vibration signals for fault detection is inherently complex, as it requires distinguishing subtle variations in frequency, amplitude, and transient behaviors that may indicate faults. The presence of overlapping frequencies and modulations further complicates the task, as distinguishing between fault-related signals and normal system behavior becomes challenging. This complexity increases the need for advanced DSP methods to effectively isolate the fault-related features from the noise and variability present in the data.

4.3 HSA-based BPF Optimization

Filtering is a primary preprocessing step that enhances the diagnostic quality of vibration data by isolating fault-related signal components. Traditional methods, like the fast kurtogram, employ static configurations that limit adaptability. However, optimization-based adaptive filtering dynamically adjusts these parameters, allowing the filter to respond to changing signal conditions. This section discusses advanced

adaptive band-pass filter (BPF) design optimized using HSA, inspired by musical harmony, iteratively adjusts filter parameters to achieve an optimal configuration that maximizes fault feature clarity. Specifically, the objective of the HSA was to optimize the three critical parameters as follows:

- **Center frequency** – The resonance frequency at which fault impacts occur.
- **Bandwidth** – The range of frequencies around the center that should be captured.
- **Filter order** – Governs the sharpness of the filter’s frequency response.

The optimization process was guided by three fitness functions: the kurtosis of the time-domain signal, the kurtosis of the ES of the vibration signal, and the sum of the SK based on the STFT. The time-domain kurtosis quantifies the “peakedness” of the vibration signal distribution, effectively highlighting impulsive fault events. The kurtosis of the ES serves as a frequency-domain indicator, capturing impulsive characteristics in the spectral content that correlate with specific fault frequencies. Finally, the STFT-based SK provides a time-frequency representation, isolating transient fault events by capturing both the intensity and precise timing of fault-related impacts, enhancing fault diagnosis through detailed temporal and frequency information.

The results concluded by the HSA-based BPF optimization are published in the following research papers: *Early-Stage Fault Diagnosis for Rotating Element Bearing Using Improved Harmony Search Algorithm with Different Fitness Functions*, published in *IEEE Transactions on Instrumentation and Measurement* and *Early-Stage Faults Detection Using Harmony Search Algorithm and STFT-Based Spectral Kurtosis*, presented and published in the conference named *Automation 2022: New Solutions and Technologies for Automation, Robotics and Measurement Techniques*.

4.3.1 Implementation of HSA

HSA was inspired by the music-based metaheuristic optimization algorithm which was first introduced in 2011 [105]. The population of the HSA consists of different musical sounds, and the aim is to find the best state of harmony. The optimization problem of a given function, subjected to the constraints, is defined below:

$$\begin{aligned} & \max f(X) \text{ or } \min f(X) & (4.1) \\ & \text{subjected to } g(X) \leq 0 \text{ and } h(X) = 0; \end{aligned}$$

where X is the set of solution candidates and the goal is to find the best solution candidate that optimizes equation (4.1), also called the fitness function.

There are three main key parameters of the HSA that are described as follows:

- Harmony Memory Size (HMS) or number of solution candidates,
- Harmony Consideration Rate (HMCR),

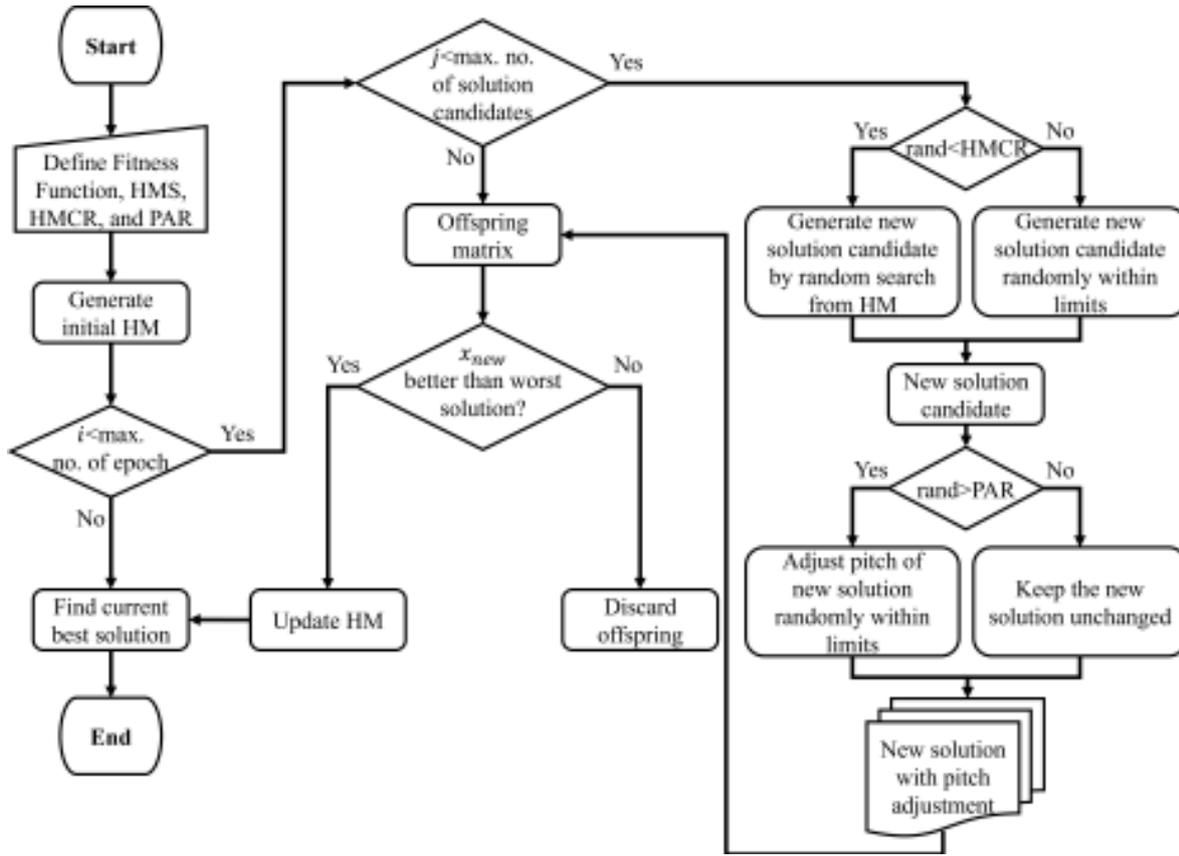


Figure 4.1: Flow diagram of proposed HSA.

- Pitch Adjustment Rate (PAR).

Generally, the values to these three key components are assigned in the following range [106] $10 \leq HMS \leq 50$, $0.7 \leq HMCR \leq 0.95$, and $0.2 \leq PAR \leq 0.5$. Figure 4.1 illustrates the overall flow diagram of the proposed HSA. The HSA consists of the following five steps.

In the first step, the specifying parameters, including the fitness function along with the constrains are defined, according to equation (4.1) and specific values are assigned to the three main parameters, i.e., HMS , $HMCR$, and PAR . In the second step, a random harmony memory (HM) of order $HMS \times N$ is generated according to the following equation:

$$HM = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_N^1 \\ x_1^2 & x_2^2 & \cdots & x_N^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{HMS} & x_2^{HMS} & \cdots & x_N^{HMS} \end{bmatrix}; \quad (4.2)$$

where, $x_i^j, i \in HMS, j \in N$ is the element of the solution candidates $x^j = x_1^j, x_2^j, x_3^j \dots x_N^j$, N is the number of elements in each solution candidate. Each element x_i^j of the solution

candidate is bounded with upper bound x_{ub} and lower bound x_{lb} as follows:

$$x_{lb} \leq x_i^j \leq x_{ub}. \quad (4.3)$$

The fitness value is calculated using equation (4.1) for each solution candidate and the worst fitness value is determined.

In the third step, a new harmony solution x_{new} is generated and its fitness value is calculated using equation (4.1). In the literature, different techniques are reported to generate the new harmony solution x_{new} . In general,

$$x_{new} = \begin{cases} x_i^j \text{ from HM with probability } HMCR \\ \text{New solution with probability } (1 - HMCR) \end{cases} ; \quad (4.4)$$

and then PAR for pitch adjustment is applied to x_{new} , as follows:

$$x_{new} = \begin{cases} x_{new} \pm bw \text{ with probability } PAR \\ x_{new} \text{ with probability } (1 - PAR) \end{cases} ; \quad (4.5)$$

where $bw = r \times (x_{ub} - x_{lb})$ and r is a positive random number such that $0 < r < 1$.

In the fourth step, the fitness values of x_{new} and the worst solution x^j from HM are compared. If the fitness value of x_{new} is better than the worst fitness value, then HM is updated by replacing the worst solution candidate x^j with x_{new} ; otherwise x_{new} is discarded.

In the fifth step, the steps three and four are repeated until the stopping criteria or termination condition are achieved. Finally, the best solution candidate is determined. This best solution candidate fulfills the optimization problem. The hyperparameters are given in Table 4.1 for the proposed HSA-based BPF.

Table 4.1: Hyperparameters of HSA.

Variable	Value	Description
N	3	number of variables
HMS	20	harmony memory size
x_{lb}	1	lower bound of variables
x_{ub}	100	upper bound of variables
$HMCR$	0.72	harmony consideration rate
PAR	0.08	pitch adjustment rate
$epoch$	50	total number of epoch

The following pseudo code outlines the step-by-step procedure of the HSA as applied to the optimization of BPF parameters.

Algorithm 1: Bandpass Filter Optimization using Improved Harmony Search

Algorithm

Input : Fitness Function, HMS, HMCR, PAR, initial HM**Output:** Bandpass Filter (central frequency, bandwidth, filter order)

```

1 // Main loop
2 for  $i \leftarrow 1$  to max. no. of epoch do
3   Construct BPF using updated HM and filter out vibration signal
4   Evaluate fitness value of each filtered signal
5   if  $j < \text{max. no. of solution candidates}$  then
6     //  $x_{new}$  generation
7     if  $\text{rand} > \text{HMCR}$  then
8       Generate  $x_{new}$  by random selection from HM
9     else
10      Generate  $x_{new}$  randomly within limits
11     // Pitch adjustment of  $x_{new}$ 
12     if  $\text{rand} > \text{PAR}$  then
13       Adjust the pitch of  $x_{new}$ 
14     else
15       Keep  $x_{new}$  unchanged
16   else
17     // Offspring's Matrix
18
19     Built offspring's Matrix
20     Construct BPF using offspring matrix and apply to vibration signal
21     Evaluate fitness value of each filtered signal
22     Accept  $x_{new}$  if better and update HM, otherwise discard  $x_{new}$ 
23   Find the current best solutions
24 // Optimized Bandpass Filter
25 Find the best solution
26 Construct the Bandpass filter
27 Filtered out the vibration signal
28 Plot ES and visualize the fault frequency

```

4.3.2 Simulation Results for HSA-based BPF

To validate the effectiveness of the HSA-based BPF, two datasets were tested: the Intelligent Maintenance System (IMS) bearing dataset and the Machinery Failure Prevention Technology (MFPT) dataset. Vibration signals from these datasets repre-

sented different fault types, including roller element and OR faults. The detail of these datasets are given in Chapter 3. Results demonstrate that HSA-based BPF produced higher SNR and clearer fault frequencies compared to the fast kurtogram-based filter, especially in noisy conditions. The simulation results are as follows:

IMS Dataset:

For the detection of Fundamental Train Frequency (FTF) faults, two vibration signals corresponding to early-stage fault conditions from IMS dataset were selected for analysis. Specifically, vibration data from file numbers 1515 and 1598 were utilized. Figure 4.2 shows the time-domain representation of the vibration signal along with its ES. The fault frequencies can be computed using the formulas given in Appendix C. Moreover, the vertical dotted lines in the figures showing these fault frequencies.

The BPF parameters were optimized using the proposed HSA, guided by three fitness functions. The resulting optimized parameters are listed in Table 4.2a. To assess the effectiveness of the proposed method, an additional BPF was designed using the fast kurtogram technique for comparison. Figure 4.3 displays the ES of the filtered signals obtained through the HSA with three fitness functions, and contrasted with the traditional fast kurtogram-based filter. The proposed method distinctly highlighted

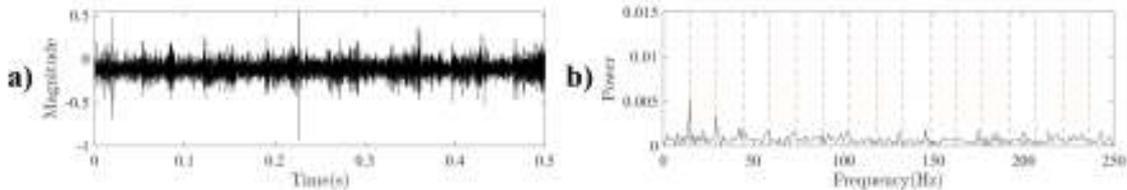


Figure 4.2: Raw vibration signal 1515 in (a) time-domain and (b) its ES from IMS bearing dataset with FTF fault.

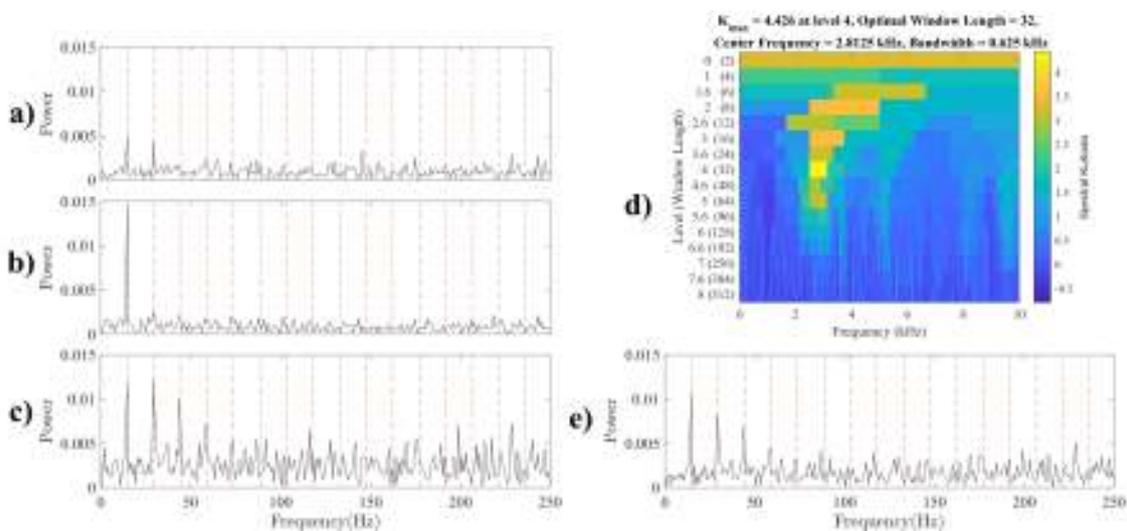


Figure 4.3: For FTF fault signal 1515; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.

Table 4.2: Filter parameters for IMS FTF faults diagnosis.

Fitness function	Center frequency	Bandwidth	Filter order
a)		1515 - FTF Fault	
First fitness function	2687.6 Hz	1118.7 Hz	46
Second fitness function	1531 Hz	843.6874 Hz	61
Third fitness function	3852 Hz	3132.3 Hz	88
b)		1598 - FTF Fault	
First fitness function	2779.6 Hz	1291 Hz	39
Second fitness function	1578.3 Hz	826.6 Hz	69
Third fitness function	5601 Hz	6110.8 Hz	59

Note: First, second, and third fitness functions are as follows:
Kurtosis of the time-domain signal, Kurtosis of the ES of the vibration signal, and Sum of the STFT-based SK.

fault frequencies with enhanced clarity, achieving SNR improvement.

Figure 4.4 shows the time-domain representation of the vibration signal along with its ES for early-stage FTF fault. For the second signal, the BPF parameters were similarly optimized using the proposed HSA, guided by the same three fitness functions.

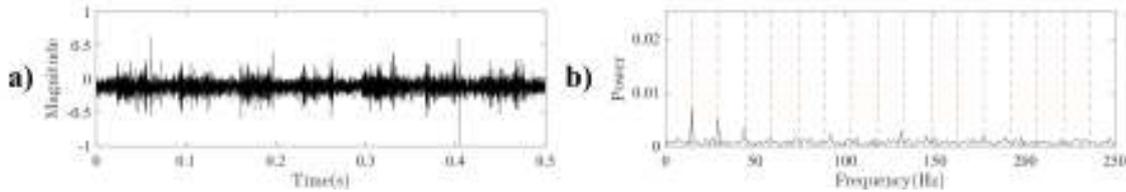


Figure 4.4: Raw vibration signal 1598 in (a) time-domain and (b) its ES from IMS bearing dataset with FTF fault.

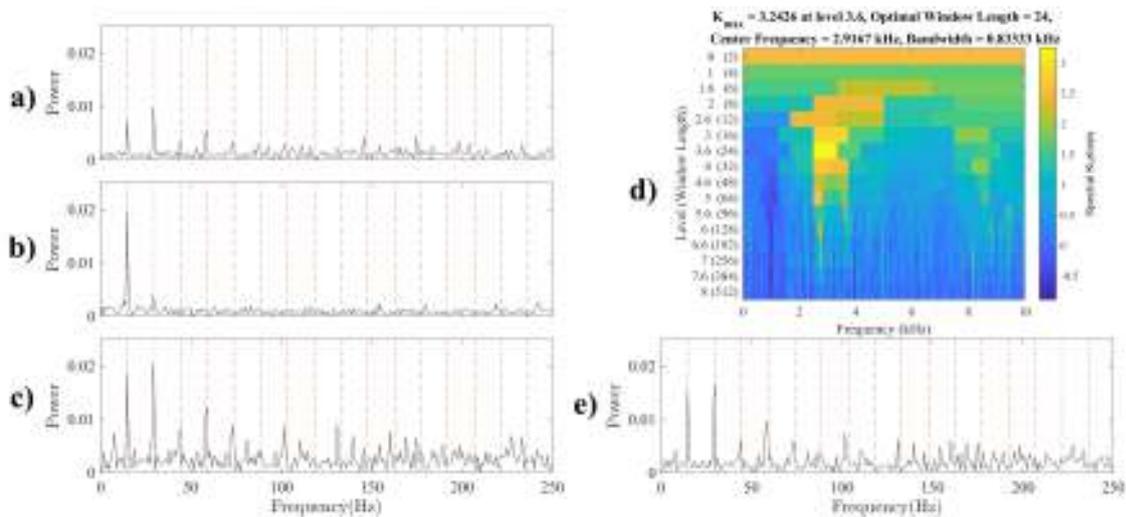


Figure 4.5: For FTF fault signal 1598; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.

The optimized parameters for this signal are presented in Table 4.2b. To evaluate the performance of the proposed approach, a comparison was also made with a BPF designed using the fast kurtogram technique. Figure 4.5 illustrates the ES of the filtered signals obtained through the HSA with the three fitness functions, alongside the results of the fast kurtogram-based filter. The proposed method effectively highlighted the fault frequencies with greater clarity, demonstrating a significant improvement in the SNR.

For the early-stage Outer Race (OR) fault analysis, vibration signals from file numbers 6075 and 6100 were utilized. Figure 4.6 illustrates the vibration signal from file 6075. The ES of the unfiltered signal did not exhibit distinct spikes corresponding to the fault frequency. The fault frequencies can be computed using the formulas given in Appendix C. Moreover, the vertical dotted lines in the figures showing these fault frequencies.

To address early-stage OR fault diagnostic, BPF were designed and optimized using the proposed HSA, guided by the same three fitness functions previously discussed. The optimized filter parameters are provided in Table 4.3a. For further validation, a BPF designed using the fast kurtogram technique was also employed for compar-

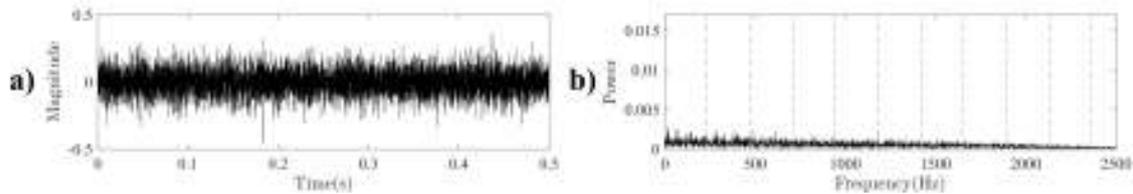


Figure 4.6: Raw vibration signal 6075 in (a) time-domain and (b) its ES from IMS bearing dataset with OR fault.

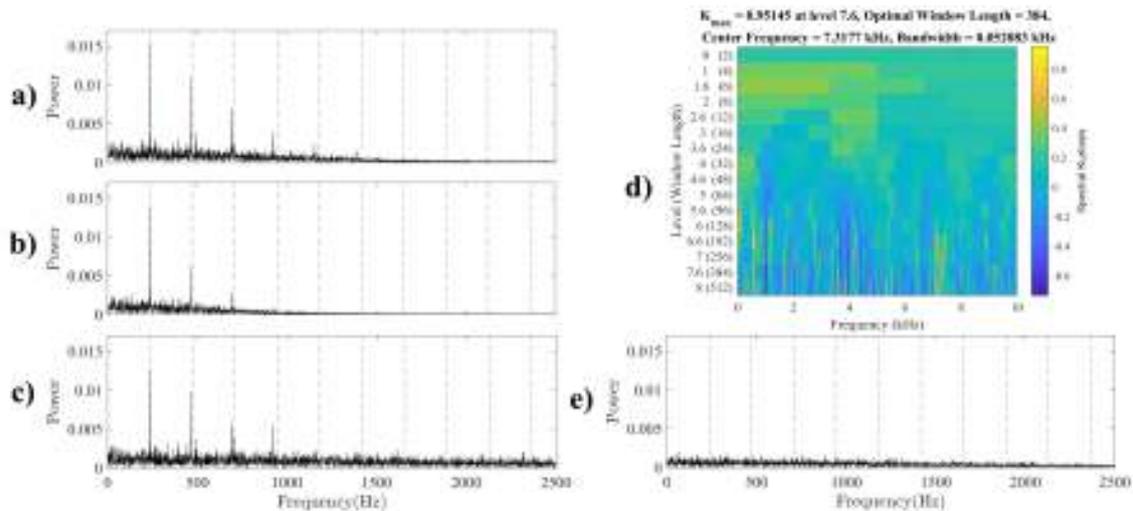


Figure 4.7: For OR fault signal 6075; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.

Table 4.3: Filter parameters for IMS OR faults diagnosis.

Fitness function	Center frequency	Bandwidth	Filter order
a)		6075 - OR Fault	
First fitness function	3917.6 Hz	2033.1 Hz	45
Second fitness function	3772.5 Hz	1288.8 Hz	53
Third fitness function	5169.3 Hz	4021.6 Hz	99
b)		6100 - OR Fault	
First fitness function	3963.2 Hz	2367.1 Hz	98
Second fitness function	3709.5 Hz	1402.1 Hz	46
Third fitness function	2943.1 Hz	4699.4 Hz	67

Note: First, second, and third fitness functions are as follows:
Kurtosis of the time-domain signal, Kurtosis of the ES of the vibration signal, and Sum of the STFT-based SK.

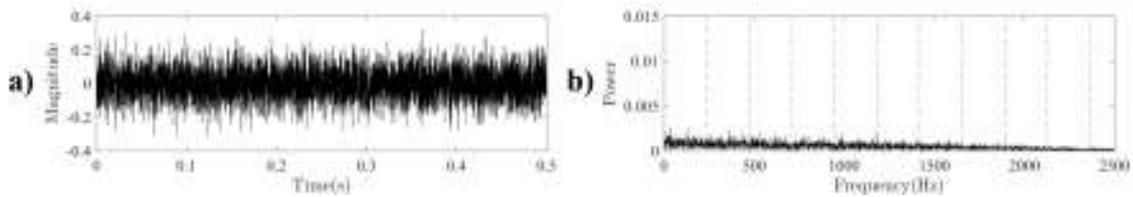


Figure 4.8: Raw vibration signal 6100 in (a) time-domain and (b) its ES from IMS bearing dataset with OR fault.

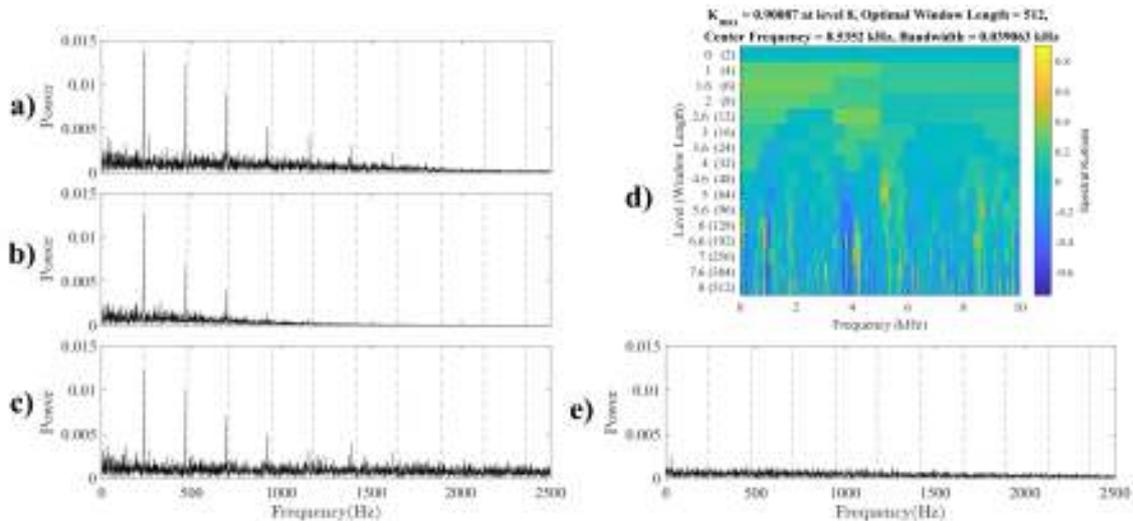


Figure 4.9: For OR fault signal 6100; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.

ison. Figure 4.7 illustrates the ES of the signals filtered using the HSA with three fitness functions, contrasted with those filtered by the fast kurtogram. The proposed method demonstrated superior clarity in highlighting fault frequencies and achieved more effective fault isolation in the frequency domain compared to traditional filtering techniques.

For the second early-stage OR fault, a vibration signal from file number 6100 was used, as depicted in Figure 4.8. The ES of the unfiltered signal did not display any clear spikes at the fault frequency. Therefore, BPF were constructed and optimized with the proposed HSA, employing the same three fitness functions discussed previously. The optimized filter parameters are also presented in Table 4.3b. To further validate the proposed method, a BPF using the fast kurtogram was again employed for comparison. Figure 4.9 illustrates the ES of the signals filtered by the HSA with three fitness functions, in comparison with those filtered by the fast kurtogram. The proposed approach effectively highlighted fault frequencies with high clarity, demonstrating superior fault isolation in the frequency domain compared to traditional filtering methods.

MFPT Dataset:

For further analysis of OR faults, two additional vibration signals from the MFPT

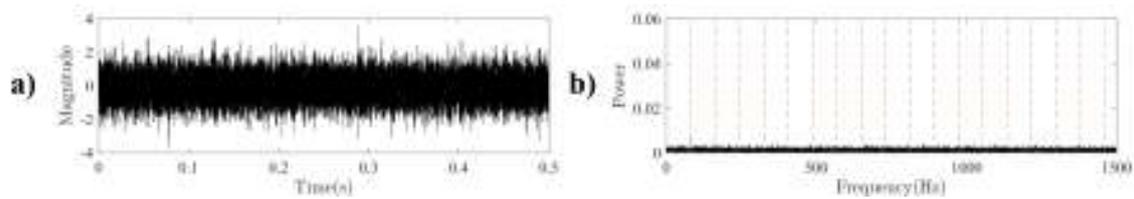


Figure 4.10: Raw vibration signal 411 in (a) time-domain and (b) its ES from MFPT dataset with OR fault.

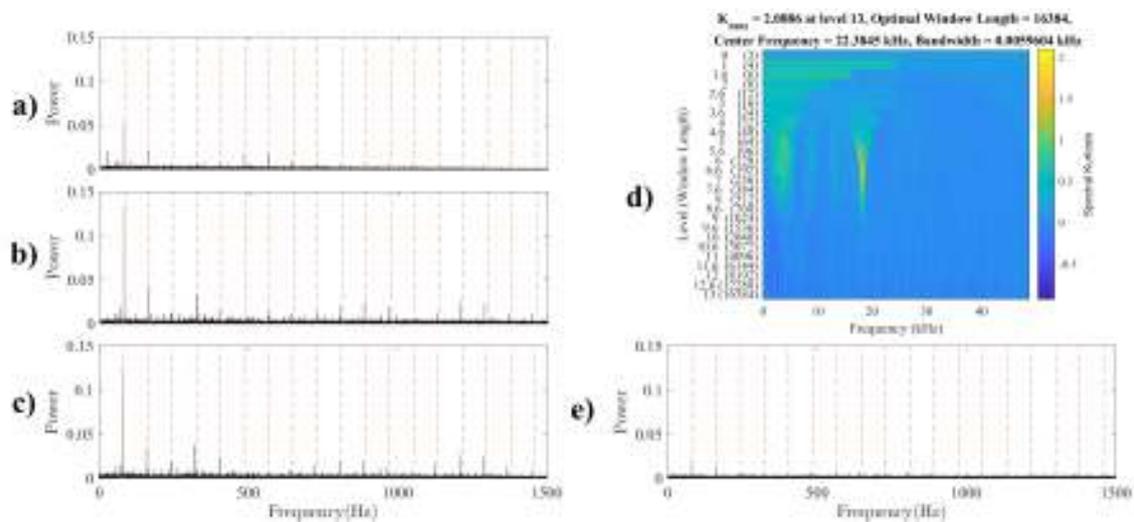


Figure 4.11: For OR fault signal 411; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.

Table 4.4: Filter parameters for MFPT bearing faults diagnosis.

Fitness function	Center frequency	Bandwidth	Filter order
a)		411 - OR Fault	
First fitness function	3854.6 Hz	1050.4 Hz	95
Second fitness function	2865 Hz	4946 Hz	60
Third fitness function	2351.9 Hz	3392.2 Hz	61
b)		413 - OR Fault	
First fitness function	10862 Hz	1377.3 Hz	99
Second fitness function	24415 Hz	48826 Hz	97
Third fitness function	6913 Hz	10218 Hz	75

Note: First, second, and third fitness functions are as follows:
 Kurtosis of the time-domain signal, Kurtosis of the ES of the
 vibration signal, and Sum of the STFT-based SK.

dataset, file numbers 411 and 413, were utilized. Figure 4.10 illustrates the time-domain and its ES for the vibration signal 411. The fault frequencies can be computed using the formulas given in Appendix C. Moreover, the vertical dotted lines in the

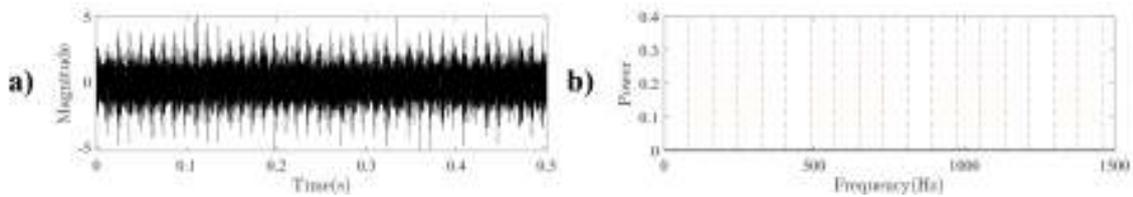


Figure 4.12: Raw vibration signal 413 in (a) time-domain and (b) its ES from MFPT dataset with OR fault.

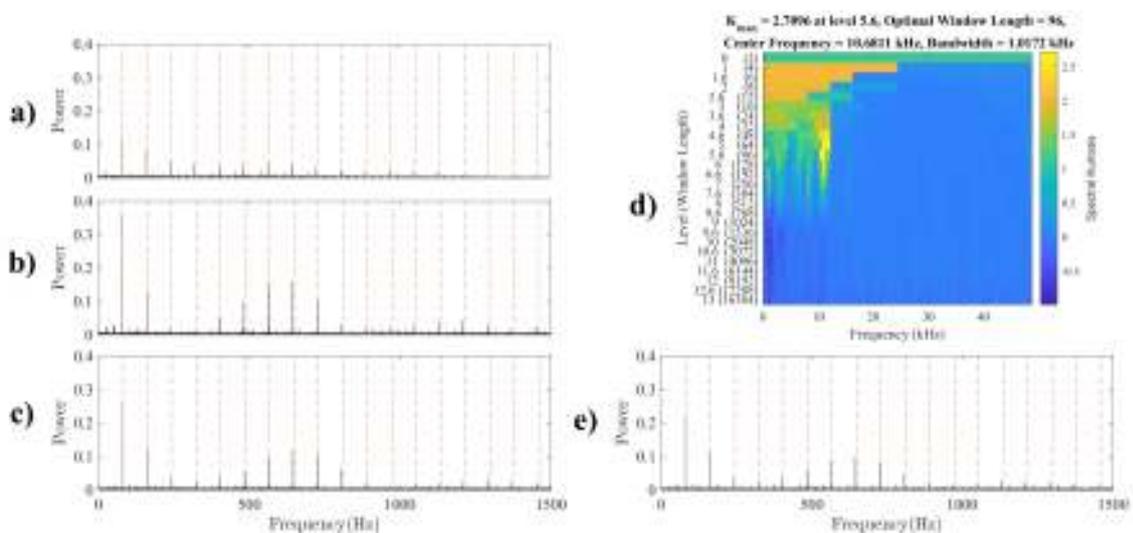


Figure 4.13: For OR fault signal 413; the resultant ES of the filtered signals with a) first fitness function, b) second fitness function, c) third fitness function, and d) fast kurtogram and e) ES of its filtered signal.

figures showing these fault frequencies. Similar to the previous approach, the bandpass filters for these signals were optimized using the proposed HS Algorithm with three fitness functions, and the optimized parameters are provided in Table 4.4a. To evaluate the proposed method's performance, a fast kurtogram-based filter was also applied to these signals for comparison. The ES of the filtered signals, as illustrated in Figure 4.11, highlights the fault frequencies with the proposed HS Algorithm method, demonstrating superior clarity and yielding an SNR improvement when compared to the fast kurtogram-based filter. The proposed approach effectively isolated fault frequencies, confirming its robustness in signal enhancement.

Similarly, for the OR fault signal from file number 413 in the MFPT dataset 4.12, the same procedure was applied. The bandpass filter was optimized using the proposed HS Algorithm guided by the three fitness functions, and the resultant optimized parameters are listed in Table 4.4b. As with previous signals, the performance of the proposed method was validated by comparing it to the fast kurtogram-based filter. Figure 4.13 shows the ES of the filtered signal, where the proposed HS Algorithm method clearly highlighted fault frequencies with high precision, offering a marked improvement in fault visibility over the traditional filtering approach. The results reinforce the efficacy of the proposed HS Algorithm model in enhancing signal clarity for fault diagnosis in OR faults.

4.4 PSO-based ARL-Wavelet BPF Optimization

Building on the successful application of the HSA-based BPF optimization, the subsequent research introduced the PSO algorithm to further refine the filter design, specifically utilizing the CWT for dynamic fault detection. The PSO is a population-based optimization algorithm inspired by the swarming behavior of animals. It adjusts filter parameters by "flying" particles (solutions) across the search space, gradually converging on an optimal configuration. In this section, PSO optimized the Asymmetric Real Laplace (ARL) wavelet, a wavelet basis function selected for its efficient frequency localization and minimal ripple effects.

The ARL-wavelet is an efficient BPF for vibration signals due to many reasons such as:

- In the frequency-domain, the ARL-wavelet has a Gaussian shape and smooth edges help to minimize the ripple effects,
- The ARL-wavelet convolution with a vibration signal retains the temporal resolution of the original signal,
- The computational time and complexity are small for the ARL-wavelet convolution with a vibration signal.

For the ARL-wavelet the sine wave is a real sine function with the frequency f_c .

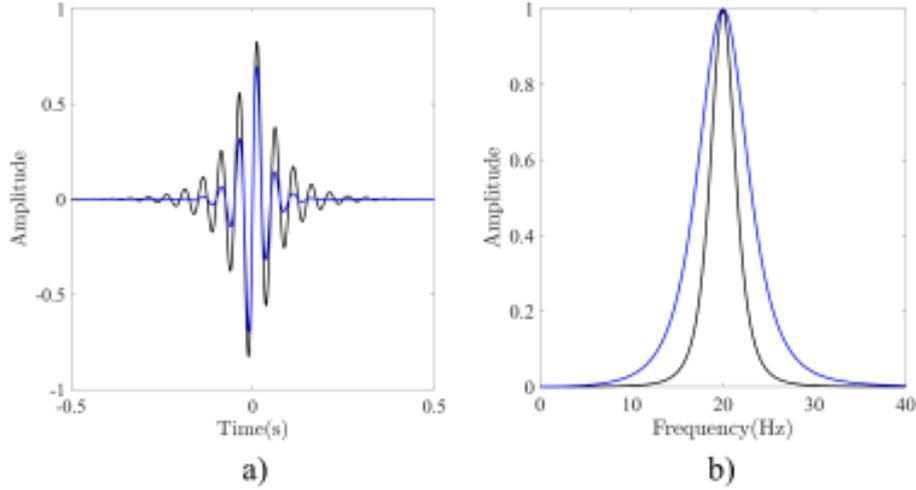


Figure 4.14: ARL-Wavelet with $f_c = 20Hz$ and $\sigma = 5Hz$ and $10Hz$ a) Time and b) frequency plots.

The wavelet function $\phi(t)$ for ARL-wavelet filter can be written as follows:

$$\phi(t) = \exp(-\pi\sigma|t|) \sin(2\pi f_c t), \quad (4.6)$$

where f_c and σ represent the center frequency and bandwidth of the ARL-wavelet-BPF. In the frequency domain, it can be written as follows:

$$\psi(f) = \frac{\sigma}{j\pi[\sigma^2 + 4(f - f_c)^2]} - \frac{\sigma}{j\pi[\sigma^2 + 4(f + f_c)^2]}, \quad (4.7)$$

Figure 4.14 illustrates the ARL-wavelet in time-domain and frequency-domain for central frequency $f_c = 20Hz$ and bandwidth $\sigma = 5Hz$ shown in black color and $\sigma = 10Hz$ representing in blue color, respectively.

The convolution of the vibration signal and the ARL-wavelet can be written as follows:

$$W_x(f_c, \sigma) = IFT[X(f)\psi^*(f)], \quad (4.8)$$

The bandwidth and center frequency of the ARL-wavelet-BPF must be selected carefully to diagnose the fault frequency in the rotating element bearing. Because of the unknown noise and fault frequency, it is challenging to find these parameters. The results concluded by the PSO-based ARL-Wavelet BPF optimization are published in the following research paper: *ARL-Wavelet-BPF Optimization using PSO Algorithm for Bearing Fault Diagnosis*, published in *Archives of Control Sciences*. The PSO optimization algorithm with SK of the ES of the vibration signal is employed in this paper to estimate these parameters.

Table 4.5: Parameters of PSO algorithm.

Variable	Value	Description
D	2	number of variables
m	50	swarm size
c_1	1.25	acceleration factor
c_2	1.75	acceleration factor
w_{min}	0.5	minimum inertia weight
w_{max}	0.9	maximum inertia weight
s_{lb}	–	lower bound of variables
s_{ub}	–	upper bound of variables
$epoch$	50	total number of epoch

4.4.1 Implementation of PSO Algorithm:

The PSO algorithm [107, 108] consists of m number of particles (the swarm size), and each particle is considered as the potential solution of the complex problem. Each particle $i \in \{1, 2, 3, \dots, k\}$ has position $P = [p_{m1}, p_{m2}, \dots, p_{mD}]$ and velocity $V = [v_{m1}, v_{m2}, \dots, v_{mD}]$ vectors of D -dimension, where D is the number of variables to be optimized in the search space. During the optimization process, the flight trajectory of the particles moves toward the optimized solution. The vectors P and V are said to be the candidate solution of the optimization problem and search direction, respectively. At each epoch, the PSO algorithm updates position and velocity vectors using the following relations:

$$\begin{aligned} v_{md}(k+1) &= Wv_{md}(k) + c_1r_1(p_{best}(k) - p_{md}(k)) + c_2r_2(g_{best}(k) - p_{md}(k)) \\ p_{md}(k+1) &= p_{md}(k) + v_{md}(k+1) \end{aligned} \quad , \quad (4.9)$$

where k is the number of epochs; W is the inertia weight associated with the previous velocity preservation; r_1 and r_2 are two random numbers in the interval $[0 \ 1]$; c_1 and c_2 are two acceleration coefficients determining the relative learning weights for p_{best} and g_{best} . $p_{md}(k)$ and $v_{md}(k)$ are the position and velocity of the m -th particle, d -th dimension at k -th epoch, respectively. Moreover, $P_{best} = [p_{best}(1), p_{best}(2), \dots, p_{best}(k)]$ is the particle best solution within each epoch also known as a historical best population while $G_{best} = [g_{best}(1), g_{best}(2), \dots, g_{best}(k)]$ is the global best solution after each epoch.

Generally, the PSO algorithm is divided into the following steps:

Step 1: In step 1, the fitness function is defined. In this research, the SK of the ES of the vibration signal is utilized as a fitness function. The value of the proposed fitness function varies with the fault frequency strength. The strength of the spikes representing fault frequency improves if the signal-to-noise ratio is enhanced. Consequently, the value of the fitness function increases and faults are diagnosed.

Step 2: In step 2, the parameters of the PSO algorithm are defined. Table 4.5

Algorithm 2: Bandpass Filter Design using Particle Swarm Optimization (PSO)

Input : Fitness Function, Population Size m , Acceleration Factors c_1, c_2 , Inertia Weight Range $[w_{\min}, w_{\max}]$, Initial Positions and Velocities

Output: Bandpass Filter (center frequency, bandwidth, filter order)

```

1 // Initialize Parameters
2 Define the fitness function (e.g., SK of the ES)
3 Set PSO parameters:  $m$  (number of particles),  $c_1, c_2$  (acceleration factors),
   $[w_{\min}, w_{\max}]$  (inertia weights), and maximum epochs
4 // Initialize Population
5 Randomly generate initial positions  $\mathbf{P} = [p_1, p_2, \dots, p_m]$  within the search space
6 Randomly initialize velocities  $\mathbf{V} = [v_1, v_2, \dots, v_m]$  within the velocity bounds
7 Evaluate the fitness of each particle and store the initial best position for each
  particle as  $\mathbf{P}_{\text{best}}$ 
8 Identify the global best position as  $\mathbf{G}_{\text{best}}$ 
9 // Main Optimization Loop
10 for  $t \leftarrow 1$  to Maximum Epochs do
11   Update inertia weight  $w$  as:
      
$$w \leftarrow w_{\max} - \frac{(w_{\max} - w_{\min})}{\text{Max Epochs}} \cdot t$$

12   for each particle  $i \leftarrow 1$  to  $m$  do
13     // Update Velocity
14     Update velocity vector:
      
$$v_i \leftarrow w \cdot v_i + c_1 \cdot r_1 \cdot (\mathbf{P}_{\text{best},i} - p_i) + c_2 \cdot r_2 \cdot (\mathbf{G}_{\text{best}} - p_i)$$

      where  $r_1, r_2$  are random values in  $[0, 1]$ 
15     // Update Position
16     Update position vector:
      
$$p_i \leftarrow p_i + v_i$$

17     Evaluate the fitness of the updated position  $p_i$ 
18     // Update Local Best
19     if  $\text{fitness}(p_i)$  is better than  $\text{fitness}(\mathbf{P}_{\text{best},i})$  then
20        $\mathbf{P}_{\text{best},i} \leftarrow p_i$ 
21     // Update Global Best
22     if  $\text{fitness}(p_i)$  is better than  $\text{fitness}(\mathbf{G}_{\text{best}})$  then
23        $\mathbf{G}_{\text{best}} \leftarrow p_i$ 
24 // Return Optimized Filter Parameters
25 Construct the bandpass filter using  $\mathbf{G}_{\text{best}}$ 
26 Apply the filter to the vibration signal
27 Plot the ES to visualize the fault frequencies

```

includes these parameters.

The lower bound and the upper bound of the variables (bandwidth and center frequency) depends on fault types. The lower bound and upper bound for bandwidth are given by $[\sigma_{lb} \ \sigma_{ub}] = [2f_f \ \frac{1}{2}f_s]$ and for the center frequency, $[f_{clb} \ f_{cub}] = [f_f \ \frac{1}{2}f_s - f_f]$, where f_f is the fault characteristic frequency which can be calculated using the equation (13), given in the next section and f_s is the sampling frequency of the vibration signal.

Step 3: In this step, the initial position and velocity vectors are generated for the whole swarm size. The size of each position and velocity matrix is $D \times m$. The values of the initial swarm are random and within the bound limits such as:

$$s_{lb} \leq s_{md} \leq s_{ub}, \quad (4.10)$$

where s_{lb} and s_{ub} are the lower bound and upper bound of the swarm and s_{md} is the generated particle.

After generating the initial swarm, the fitness value of each potential solution is calculated using the proposed fitness function described above. The initial p_{best} is determined using the fitness values of each potential solution where the potential solution with the maximum fitness value represents the initial g_{best} .

Step 4: In step 4, p_{best} and g_{best} are updated. For that, the initial or previous p_{best} is compared with the current p_{best} evaluated by the updated swarm. The maximum fitness value of the resultant p_{best} represents the current g_{best} which is compared to the previous g_{best} to determine the resultant g_{best} in each epoch.

Step 5: In step 5, the position and velocity vectors are updated using the equation (4.9). The updated position and velocity must be within the bound limits according to the equation (4.10).

Step 6: In the final step, the optimized value among the G_{best} vector is determined. This g_{best} value corresponds to the maximum fitness value and determines the optimized parameters for the proposed Morlet wavelet filter for the faulty vibration signal.

4.4.2 Simulation Results for PSO-based ARL-Wavelet BPF

For validation, the PSO-based ARL-wavelet BPF was tested on MFPT bearing signals with known fault frequencies. The results demonstrated that PSO optimization provided a substantial increase in SNR, enhancing fault frequency visibility in the ES compared to unfiltered signals. Figure 4.15 and Figure 4.16 illustrate the OR faults in bearings. Whereas, Figure 4.17 and Figure 4.18 show the baseline fault frequency conditions. The fault frequencies can be computed using the formulas given in Appendix C. Moreover, the vertical dotted lines in the figures showing these fault frequencies. All these experimental results depict the unfiltered and filtered time-domain vibration signals and their ES. The results show that the performance of the proposed PSO

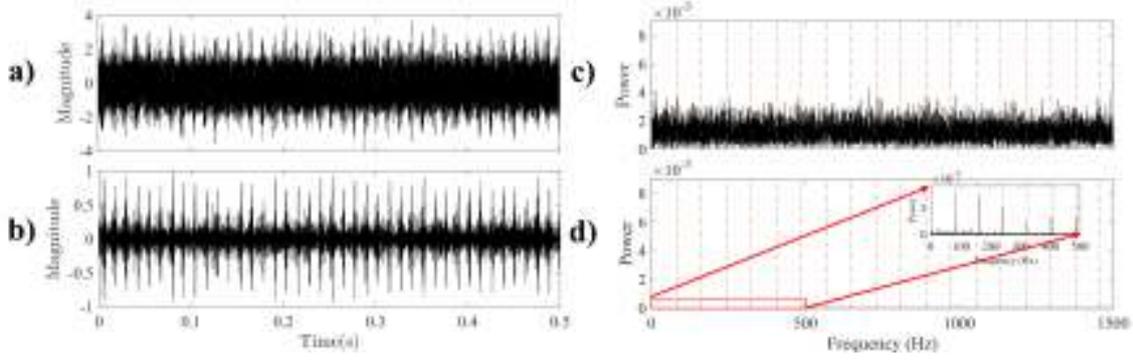


Figure 4.15: First OR fault from MFPT dataset: a) time-domain vibration signal, b) filtered time-domain signal, c) ES of unfiltered signal, and d) ES of filtered signal.

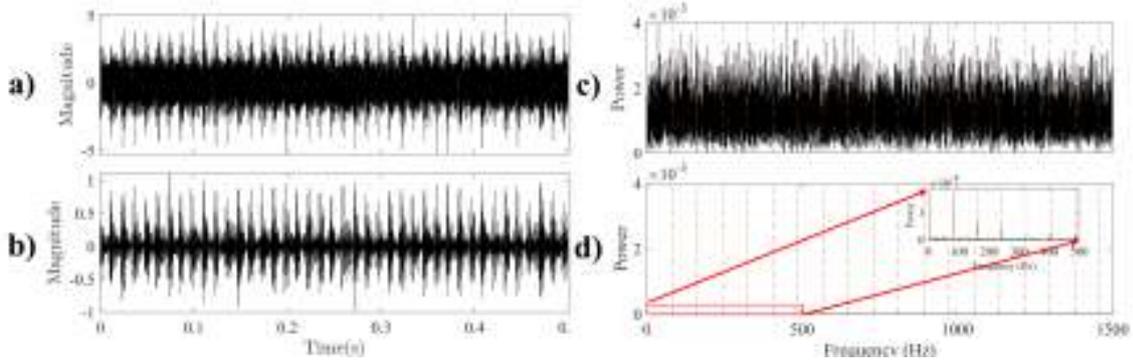


Figure 4.16: Second OR fault from MFPT dataset: a) time-domain vibration signal, b) filtered time-domain signal, c) ES of unfiltered signal, and d) ES of filtered signal.

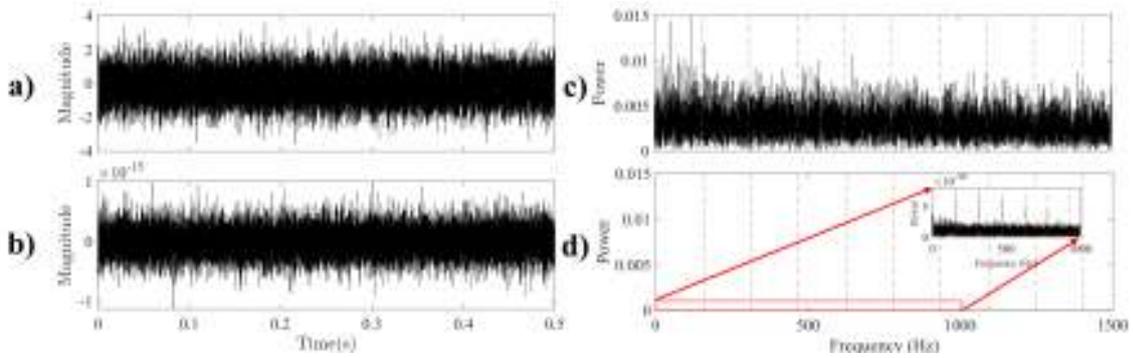


Figure 4.17: First baseline fault from MFPT dataset: a) time-domain vibration signal, b) filtered time-domain signal, c) ES of unfiltered signal, and d) ES of filtered signal.

algorithm method is efficient to diagnose the fault frequencies with high SNR.

Furthermore, SNR is also calculated for all the experiments for quantitative comparison. Table 4.6 shows the relationship between SNR for unfiltered signals and filtered signals. For the filtered signal, SNR is calculated using the following formula:

$$\text{SNR}_{\text{filtered signal}} = \frac{\sum_{n=1}^s \max_{n \in V_{\text{peaks}}} (ES_f[n]^2) / s}{\sum_{n=1}^l ES_s[n]^2 / l}, \quad (4.11)$$

where s is the number of spikes in the ES of the filtered signal that is taken to compute

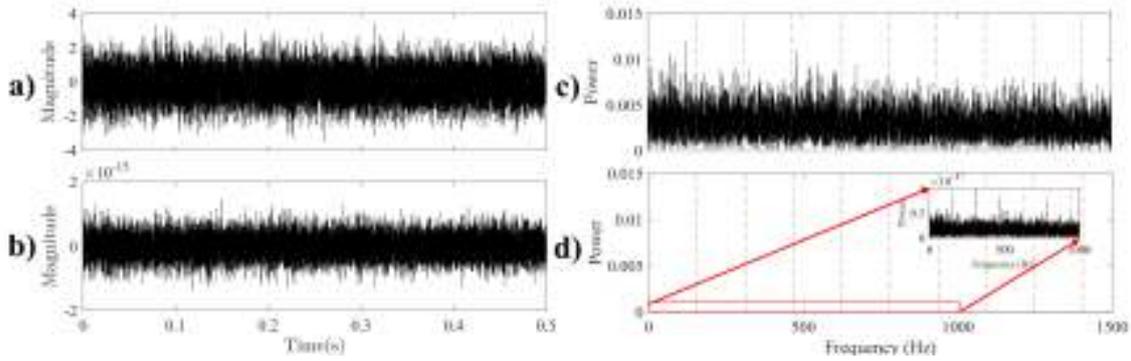


Figure 4.18: Second baseline fault from MFPT dataset: a) time-domain vibration signal, b) filtered time-domain signal, c) ES of unfiltered signal, and d) ES of filtered signal.

the SNR, V_{peaks} is a vector containing all the spikes, $ES_f[n]$ is the ES of the filtered signal. l is the number of spikes in the ES of the unfiltered signal, and $ES_s[n]$ is the ES of the unfiltered signal. Note that we have considered the whole unfiltered vibration signal to calculate the SNR, hence, l is equal to the total number of samples in the ES of the unfiltered vibration signal.

Table 4.6: Quantitative comparison of SNR.

Experiment	SNR	SNR	SNR	SNR
	<i>unfiltered signal</i>	<i>filtered signal</i>	db_u	db_f
1	4.4814	96.3686	6.5141	19.8394
2	3.7846	155.2415	5.7802	21.9101
3	2.9334	19.1820	4.6737	12.8289
4	1.9944	19.9933	2.9982	13.0089

To calculate the SNR for an unfiltered signal, the same number of spikes are taken from the ES of the vibration signal with the same position as the ES of the filtered signal.

4.5 IMF-based Vehicle Engine Misfire Diagnostics

The identification of engine misfires is a critical diagnostic task, as misfires disrupt the combustion cycle, leading to decreased performance, higher emissions, and potential long-term damage to engine components. Vibration signal analysis provides a non-intrusive method for diagnosing misfires by examining the unique patterns these events generate. This section explores the frequency-domain analysis techniques used in this study, specifically FFT, ES, and EMD. These methods were applied to engine vibration signals captured under varying RPMs and load conditions, providing a comprehensive approach to misfire detection.

The vibration signal $x(t)$ is decomposed into IMFs $\{IMF_1(t), IMF_2(t), \dots, IMF_k(t)\}$ and a residue $r(t)$ as follows:

$$x(t) = \sum_{i=1}^k IMF_i(t) + r(t)$$

where each IMF represents an oscillatory mode with different misfire frequency characteristics. The process of extracting each IMF involves the steps outlined above, and the residue represents the remaining vibration signal after extracting all the IMFs. This iterative process allows EMD to decompose complex, non-linear, and non-stationary vibration signals into simpler, intrinsic oscillatory modes.

The vibration data was collected using an ADXL1002 MEMS accelerometer interfaced with a Beaglebone Black microcontroller, capturing engine vibrations under different operating states: unloaded, loaded, and misfire conditions at RPMs of 1500, 2500, and 3000. The ADXL1002, chosen for its high sensitivity and frequency range, was calibrated to ensure accurate data acquisition, as described in Chapter 2.

The results concluded by the IMF-based vehicle engine misfire diagnostics using low-cost ADXL1002 accelerometer are published in the following research paper: *Analysis of the Vehicle Engine Misfires using Frequency-Domain Approaches at Various RPMs with ADXL1002 Accelerometer*, published in *Archives of Acoustics*.

4.5.1 Implementation of IMF

EMD is a data-driven method that decomposes a complex vibration signal into a set of IMFs. These IMFs represent different oscillatory modes of the signal with well-defined misfire frequency content. The process of EMD is based on the principle of extracting oscillatory modes by iteratively sifting the signal through envelopes of local maxima and minima.

The original vibration signal $x(t)$ undergoes a series of steps to compute the IMFs. Initially, the signal $x(t)$ is set as the residue, which forms the starting point for the decomposition process:

$$\text{residue} = x(t)$$

The set of IMFs is initialized as an empty collection:

$$IMFs = \emptyset$$

The iterative process begins by checking the residue for monotonicity. The algorithm proceeds until the residue becomes monotonic, meaning it has fewer than two extrema.

This condition for monotonicity is expressed as:

$$\text{Monotonic}(r(t)) = (\text{Number of Extrema of } r(t)) < 2$$

where $r(t)$ refers to the residue at each iteration.

During each iteration of the sifting process, the oscillatory component of the signal is extracted by subtracting the mean envelope. To achieve this, the local maxima and minima of the current residue $r(t)$ are identified. The positions and values of these extrema, denoted as $\max(t_i)$ and $\min(t_i)$, are determined:

$$\max(r(t)) \quad \text{and} \quad \min(r(t))$$

Next, cubic spline interpolation is used to construct the upper and lower envelopes from the local maxima and minima:

$$\text{upperEnvelope}(t) = \text{interp1}(\text{maxLocs}, \text{maxPeaks}, t, \text{'spline'})$$

$$\text{lowerEnvelope}(t) = \text{interp1}(\text{minLocs}, \text{minPeaks}, t, \text{'spline'})$$

The mean envelope is then computed as the average of the upper and lower envelopes:

$$\text{meanEnvelope}(t) = \frac{\text{upperEnvelope}(t) + \text{lowerEnvelope}(t)}{2}$$

Once the mean envelope is determined, the current signal detail is updated by subtracting the mean envelope from the residue:

$$\text{detail}(t) = r(t) - \text{meanEnvelope}(t)$$

To ensure convergence, the difference between the newly computed detail and the previous detail is checked. If the maximum absolute difference is smaller than a specified threshold tol , convergence is considered to have been reached:

$$\max(|\text{detail}(t) - \text{previousDetail}(t)|) < \text{tol}$$

The extracted detail is stored as one of the IMFs:

$$\text{IMFs} = [\text{IMFs}; \text{detail}]$$

The residue is then updated by subtracting the extracted IMF from the current residue:

$$\text{residue}(t) = \text{residue}(t) - \text{detail}(t)$$

The process continues until either the residue becomes monotonic or all the IMFs have been extracted. The pseudo code the whole procedure to compute IMF is given in Algorithm 3.

Algorithm 3: Empirical Mode Decomposition (EMD) Algorithm

Input : signal, maxSift, tol
Output: IMFs, residue
Data: signal: Input vibration signal
Data: maxSift: Maximum number of sifting iterations for each IMF
Data: tol: Convergence threshold for sifting

```

1 residue ← signal
2 IMFs ← []
3 while not isMonotonic(residue) do
4     detail ← residue
5     for iter ← 1 to maxSift do
6         [maxPeaks, maxLocs] ← findpeaks(detail)
7         [minPeaks, minLocs] ← findpeaks(-detail)
8         minPeaks ← -minPeaks
9         if length(maxLocs) < 2 or length(minLocs) < 2 then
10            break
11        upperEnvelope ← interp1(maxLocs, maxPeaks, 1 ... length(signal),
            'spline')
12        lowerEnvelope ← interp1(minLocs, minPeaks, 1 ... length(signal),
            'spline')
13        meanEnvelope ← (upperEnvelope + lowerEnvelope) / 2
14        previousDetail ← detail
15        detail ← detail - meanEnvelope
16        if max(abs(detail - previousDetail)) < tol then
17            break
18    IMFs ← append(IMFs, detail)
19    residue ← residue - detail
20 return IMFs, residue

```

4.5.2 Simulation Results for IMF-based Misfire Diagnostics

FFT Analysis

In analyzing signals at 1500 RPM, 2500 RPM, and 3000 RPM, the FFT revealed harmonic frequencies associated with normal engine operation. However, in misfire conditions, there was a notable reduction in amplitude at the expected firing frequency

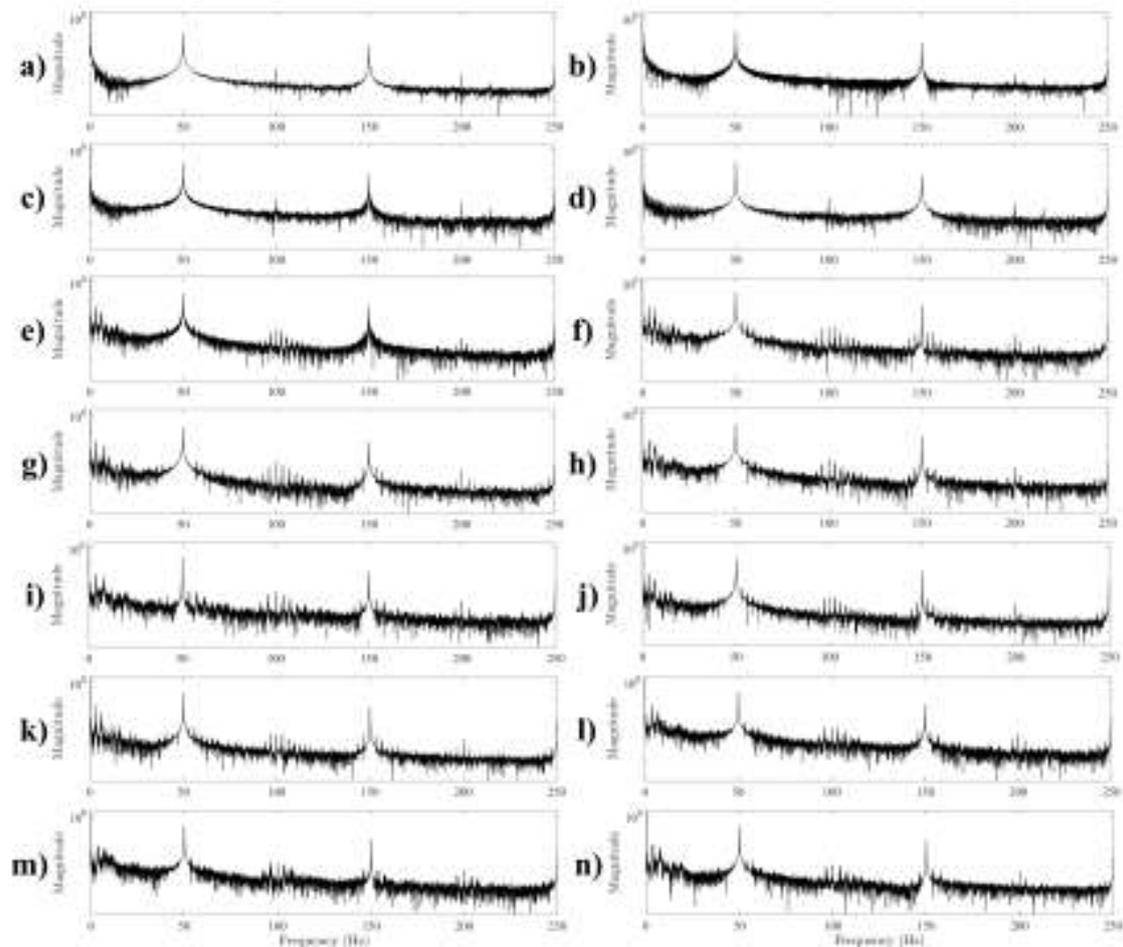


Figure 4.19: FFT of the vibration signals recorded from the vehicle engine using ADXL1002 accelerometer interfaced with BBB: a & b) FFT of the healthy signals without load and misfire conditions, c, d, e, f, g, & h) FFT of the healthy signals with load and without misfire conditions, and i, j, k, l, m, & n) FFT of the faulty signals with load and with misfire conditions.

(e.g., 25 Hz for 1500 RPM), with irregular spectral content around harmonic frequencies. These variations, shown in Figure 4.19, indicated that misfires alter the spectral distribution, yet FFT alone was insufficient for isolating transient misfire components due to overlapping frequencies and noise [100].

ES Analysis

To address the limitations of DFT, the ES was applied to the vibration signals. ES involves calculating the amplitude envelope of the signal and examining its frequency content, effectively highlighting periodic modulations induced by load and misfire events.

The ES analysis was conducted under both loaded and misfire conditions across varying RPMs. The ES for healthy condition of the vehicle engine is shown in Figure 4.20. The figure shows no spikes because of healthy conditions under no load, half load and full load conditions. Whereas, for misfire conditions at various RPMs, for exam-

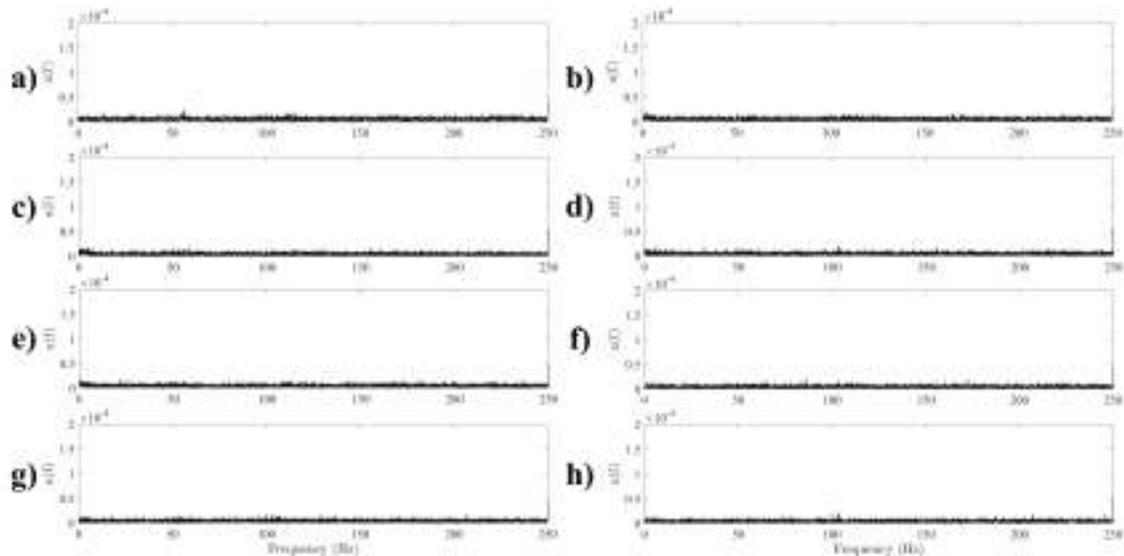


Figure 4.20: ES of the healthy vibration signals recorded from the vehicle engine using ADXL1002 accelerometer interfaced with BBB: a) ES of the healthy signal recorded at 1500 RPMs (25 Hz) with no load, b) ES of the healthy signal recorded at 3000 RPMs (50 Hz) with no load, c & d) ES of the healthy signals recorded at 1500 RPMs (25 Hz) with misfire conditions at half load and full load, e & f) ES of the healthy signals recorded at 2500 RPMs (41.67 Hz) with misfire conditions at half load and full load, and g & h) ES of the healthy signals recorded at 3000 RPMs (50 Hz) with misfire conditions at half load and full load.

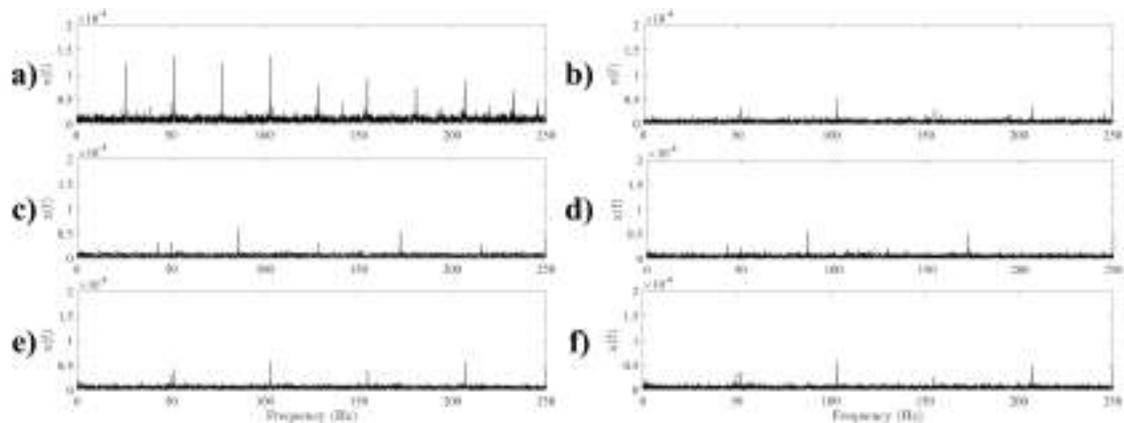


Figure 4.21: ES of the misfired vibration signals recorded from the vehicle engine using ADXL1002 accelerometer interfaced with BBB: a & b) ES of the faulty signals recorded at 1500 RPMs (25 Hz) with misfire conditions at half load and full load, c & d) ES of the faulty signals recorded at 2500 RPMs (41.67 Hz) with misfire conditions at half load and full load, and e & f) ES of the faulty signals recorded at 3000 RPMs (50 Hz) with misfire conditions at half load and full load.

ple, at 1500 RPM with a misfire, ES revealed prominent peaks at 25 Hz (the misfire frequency), which were more distinct than in the FFT results. Figure 4.21 displays these results at different loads, clearly indicating that misfire frequencies appear as well-defined spikes in the ES. At 2500 RPM and 3000 RPM, ES analysis successfully

isolated misfire frequencies at 41.67 Hz and 50 Hz, respectively, under different load conditions. These results validated the effectiveness of ES in detecting misfire events, as it suppressed noise and enhanced the clarity of fault-induced frequency components [100]. However, these spikes does not shows the other side frequencies occurred because of load conditions along with misfire. Therefor, the recorded signals needs to be refined from unwanted noises and other frequencies.

EMD and IMF Analysis

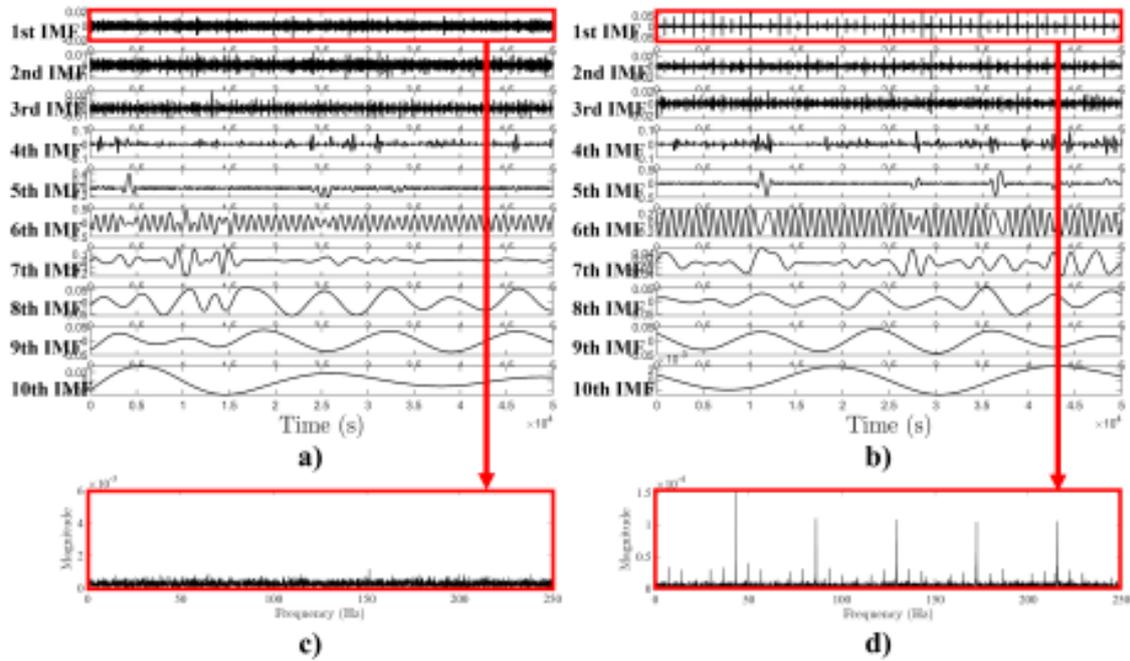


Figure 4.22: Comparison of healthy and misfire signals using FFT of the first-IMF: a) EMD of the healthy vibration signal with load at 2500 RPMs, b) EMD of the misfire vibration signal with load at 2500 RPMs, c) FFT of the first-IMF from (a), and d) FFT of the first-IMF from (b).

To further refine the signal analysis, EMD was applied. EMD is a time-frequency technique that decomposes non-linear and non-stationary signals into IMFs as described above in this chapter, each representing a distinct oscillatory mode within the signal. The decomposition process involves iterative sifting, where upper and lower signal envelopes are calculated, and their mean is subtracted from the original signal to derive each IMF. This adaptive approach is particularly effective for transient fault detection, as each IMF captures oscillations at different frequency scales, revealing fault-related components masked in raw data.

Figure 4.23 illustrates the FFT of first-IMF for healthy signals at different RPMs and load conditions. This figure show no spikes at any frequency because of the healthy condition of the vehicle engine. To compare the healthy and misfire conditions, Figure 4.22 illustrates the EMD results for signals recorded at 2500 RPM and 3000 RPM under

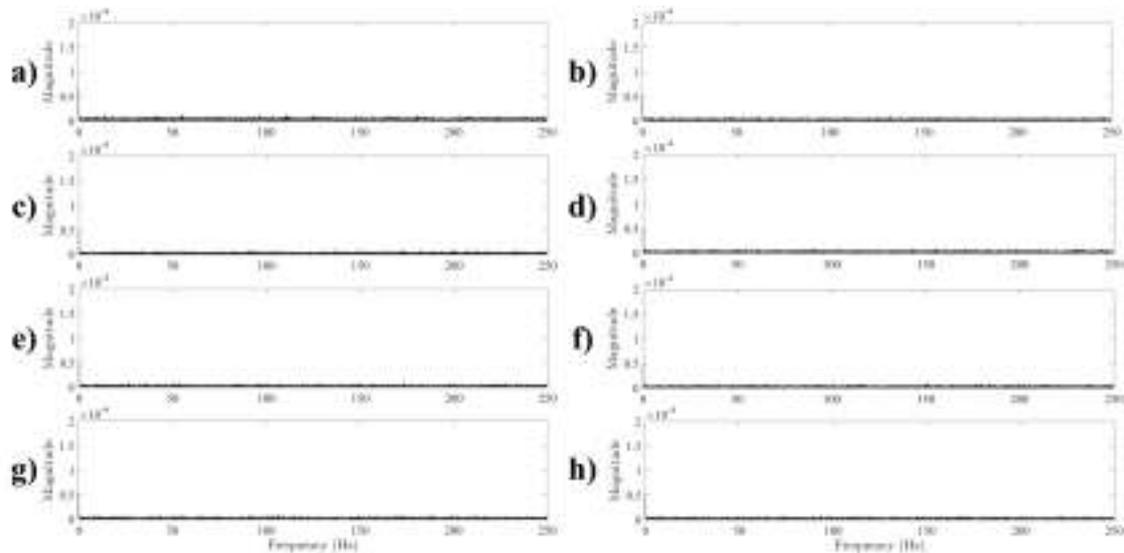


Figure 4.23: FFT of the first-IMF: a & b) healthy signals with 1500 (25 Hz) and 3000 RPMs (50 Hz) without misfire conditions and unload, c & d) healthy signals with 1500 RPMs (25 Hz) without misfire conditions at half load and full load, e & f) healthy signals with 2500 RPMs (41.67 Hz) without misfire conditions at half load and full load, and g & h) healthy signals with 3000 RPMs (50 Hz) without misfire conditions at half load and full load.

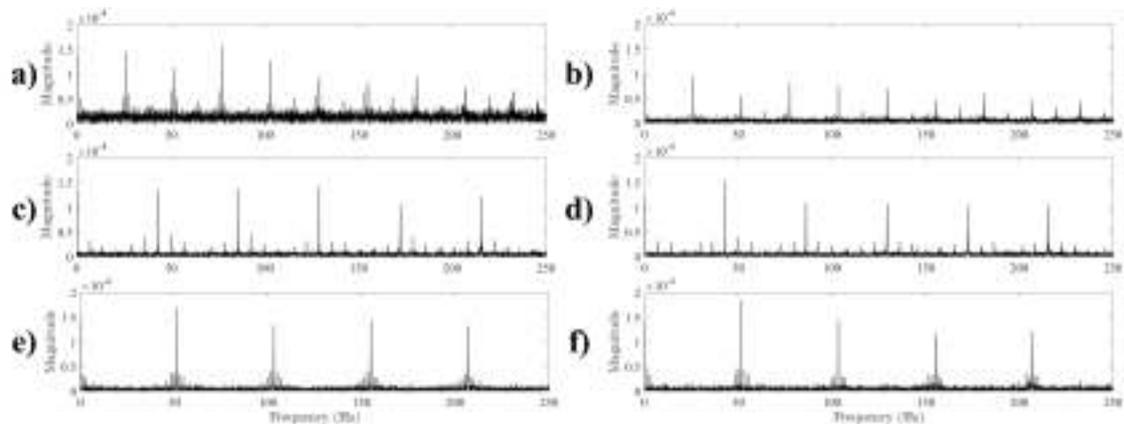


Figure 4.24: FFT of the first-IMF: a & b) faulty signals with 1500 RPMs (25 Hz) and with misfire conditions at half load and full load, c & d) faulty signals with 2500 RPMs (41.67 Hz) and with misfire conditions at half load and full load, and e & f) faulty signals with 3000 RPMs (50 Hz) and with misfire conditions at half load and full load.

loaded conditions with and without misfire. The first-IMF contained high-frequency components where fault-induced impacts are most apparent. FFT analysis on the first-IMF at 1500 RPM revealed a misfire frequency peak at 25 Hz with clear harmonic structures, indicating that EMD successfully isolated the misfire's transient signatures from background noise. Similar patterns were observed at 2500 RPM (41.67 Hz) and 3000 RPM (50 Hz) across both half and full load conditions (Fig. 4.24). These findings demonstrate that EMD is an effective tool for identifying misfires by extracting low-frequency fault information from high-frequency noise [?]Ahsan2024aoa). Moreover,

these methods formed a comprehensive approach to vehicle misfire diagnosis, with ES and IMF-based approach showing particular strength in misfire frequencies detection [100]. This layered analysis approach improves diagnostic reliability, enabling accurate misfire identification under various operational conditions.

4.6 Comparative Analysis of DSP Methods

This section provides a comparative analysis of the DSP methods applied to both bearing datasets and vehicle engine vibration datasets, integrating insights from the previously discussed filtering and signal analysis methods. By evaluating the performance, adaptability, and diagnostic accuracy of each technique, the analysis highlights their effectiveness in addressing the complexities of engine vibration data and bearing signals, particularly under diverse operational conditions. This comprehensive approach aims to identify the most suitable methods for diagnosing faults and enhancing the reliability of vehicle systems.

4.6.1 Diagnostic Performance of Filter Design Approaches

In Sections 4.3 and 4.4, adaptive filter optimization based on HSA and PSO algorithms demonstrated significant improvements in the SNR for early-stage fault diagnosis in bearing datasets. Specifically, HSA-based BPF optimization, using kurtosis, SK, and STFT-based SK as fitness functions, effectively isolated early-stage fault frequencies in bearing signals. This configuration provided adaptability by dynamically adjusting to relevant frequency bands, an essential feature for detecting transient misfire events in non-stationary environments.

The PSO algorithm, used with the CWT, further enhanced diagnostic precision by fine-tuning the Asymmetric Real Laplace (ARL) wavelet parameters. This approach yielded substantial SNR improvements, especially in low-SNR environments, allowing critical fault components to emerge more clearly in the ES.

4.6.2 Diagnostic Performance of FFT, ES, and EMD Approaches

The diagnostic analysis in Section 4.5 employed FFT, ES, and EMD methods to identify misfire frequencies within the vehicle engine vibration dataset across varying engine speeds and load conditions. This dataset, acquired using the cost-effective ADXL1002 MEMS accelerometer interfaced with the BeagleBone Black, as described in Chapter 2, served as the foundation for the analysis. The initial application of FFT directly to the raw vibration data proved ineffective for diagnosing misfire frequencies due to the presence of extraneous noise and non-stationary components. Similarly, while ES applied to the raw vibration dataset successfully extracted fault frequencies

associated with misfire, it consistently revealed an unidentified frequency at 50 Hz across all signals, indicating the presence of external noise or interference.

To address this, EMD was applied to decompose the signal, isolating and removing unwanted noise and artifacts. Subsequently, FFT was performed on the first-IMF extracted by EMD, enabling the precise identification of fault frequencies associated with engine misfire. Each frequency-domain technique demonstrated unique strengths in addressing various complexities of the vibration data, contributing significantly to the accuracy and robustness of the misfire detection process.

- **FFT:** As an initial frequency-domain tool, FFT provided a baseline view of the engine's harmonic frequencies, which is useful for general vibration analysis. However, FFT alone struggled with transient detection due to frequency overlap and noise interference, particularly at higher engine speeds and under misfire conditions. Consequently, while FFT offers quick frequency insights, its limitations in handling transient signals make it less reliable for accurate misfire detection in isolation.
- **ES:** ES analysis excelled at identifying amplitude modulations linked to misfire events, effectively distinguishing fault frequencies even under noisy conditions. ES revealed fault-specific frequency peaks, such as 25 Hz at 1500 RPM, which were obscured in the FFT results. This precision makes ES a valuable tool for enhancing the clarity of misfire-related frequencies, especially in the presence of high noise levels. The ES was particularly effective for diagnostic applications that require sensitivity to fault-induced amplitude fluctuations, as it provides a focused spectral representation of these events.
- **EMD:** EMD demonstrated strong capabilities in isolating transient fault components by decomposing the signal into IMFs, which represent oscillatory modes across different frequency bands. This decomposition allowed for further FFT analysis on individual IMFs, where the first-IMF consistently highlighted misfire frequencies that were masked in the raw data. For instance, the first-IMF of the misfire data at 1500 RPM clearly indicated the expected 25 Hz frequency with associated harmonic structures, while subsequent IMFs captured other frequency components related to normal operation. This ability to separate complex, overlapping frequencies makes EMD a powerful tool for identifying transient misfire events within engine vibration signals.

4.7 Conclusion

This chapter provided a comprehensive analysis of advanced DSP techniques for fault diagnosis using both bearing datasets and vehicle engine misfire datasets. By applying these methods to diverse and complex vibration signals, the study demonstrated

their effectiveness in isolating fault-specific features, whether related to bearing defects or engine misfires. The results emphasized the adaptability of DSP approaches in handling noisy and intricate datasets, showcasing their potential to enhance diagnostic accuracy and reliability across different fault scenarios. This dual focus highlights the versatility of DSP tools in addressing a wide range of vehicle diagnostics challenges.

The investigation into filter design underscored the efficacy of adaptive optimization approaches, such as the HSA and PSO algorithms. These algorithms tailored filtering parameters to dynamically target fault-related frequency bands, demonstrating superior SNR improvements over traditional methods. While HSA excelled in SK-based configurations for bearing fault analysis, PSO, in conjunction with the CWT, emerged as a robust framework for transient fault detection. These methodologies are well-suited for application in early-stage faults in bearing, where adaptability to varying operational conditions is critical.

The chapter also evaluated signal analysis techniques—FFT, ES, and EMD—to diagnose engine misfires across different speeds and loads. FFT provided an essential baseline for frequency-domain insights but was ineffective in isolating transient events. ES effectively highlighted amplitude modulations, distinguishing misfire frequencies with remarkable clarity under noisy conditions. EMD, with its ability to decompose non-stationary signals into IMFs, excelled at revealing transient fault components, making it a powerful complement to FFT for detailed analysis.

In synthesizing these techniques, the chapter demonstrated the importance of integrating multiple DSP methods to achieve comprehensive fault detection. The combination of adaptive filtering with advanced signal decomposition and spectral analysis offers a robust diagnostic framework, capable of addressing the complexities of real-world vibration data. This holistic approach not only enhances the precision of misfire detection but also provides a scalable solution for broader fault diagnosis applications in automotive engineering.

Chapter 5

Artificial Intelligence Methods for Vehicle Diagnostics

5.1 Introduction

The development and integration of artificial intelligence (AI) in the field of fault diagnosis has transformed how complex mechanical systems are monitored, analyzed, and maintained [31, 109, 110, 111, 112]. Traditional diagnostic methods, though effective in detecting various faults, often face limitations in processing large and complex datasets and in adapting to varying operating conditions. In response to these limitations, AI has emerged as a robust solution, enabling automated feature extraction, precise pattern recognition, and adaptive learning, all of which enhance the fault diagnosis process. Among these emerging applications, fault diagnosis in internal combustion (IC) engines, specifically the detection of misfire events through vibration signals, has become a significant focus within the field.

Misfire events in IC engines, characterized by an incomplete combustion cycle, lead to engine inefficiency, increased emissions, and potential damage over time. Prompt and accurate detection of these events is essential, particularly in the automotive industry, where regulations around emissions and fuel efficiency are becoming increasingly stringent. Vibration signals, which naturally arise from the mechanical components of an engine, are a valuable source of information for diagnosing such faults [98, 96, 19, 113]. However, due to their complex and non-stationary nature, these signals require advanced signal processing and pattern recognition techniques to extract meaningful information about engine health [114, 115, 116, 117]. While traditional approaches rely on expert-defined features from these signals, AI-based methods offer the advantage of automatically learning and extracting relevant features, making them highly suitable for such a nuanced task [118, 119].

This chapter explores the application of Deep Convolutional Neural Networks (DCNN)

and hybrid model of DCNN-Long Short-Term Memory (DCNN-LSTM) to diagnose engine misfire using vibration data, a complex and dynamic type of mechanical fault that often eludes traditional diagnostic methods. DCNN is a powerful architecture and is especially effective for image-like data due to its layered structure, which identifies patterns hierarchically, from low-level features like edges to higher-level, more abstract features [120, 121, 122, 123, 124]. In this study, both one-dimensional (1D) and two-dimensional (2D) representations of vibration signals were used to fully leverage CNN's capabilities in detecting misfire signatures.

Furthermore, to improve diagnostic accuracy and capture temporal dependencies in vibration data, a hybrid model combining DCNN and LSTM units was employed. While CNNs excel in capturing spatial features in data, they lack the ability to learn from sequences, which is where LSTMs—designed to retain information over time—become valuable. The DCNN-LSTM hybrid architecture thus combines the strengths of both models, enabling the capture of both spatial and temporal patterns within vibration signals. This hybrid model approach aligns well with the objective of this study: to develop a reliable and robust diagnostic framework that can identify engine misfires with high accuracy, even under varied operating conditions. Moreover, hybrid models are further improved using the multi-Kernel approach for both 1D and 2D datasets that show the superiority over the hybrid models.

This chapter details the implementation of the AI methods for vehicle misfire detection. It begins with an overview of the AI techniques employed. This is followed by a detailed discussion of the vehicle dataset and the preprocessing steps undertaken to create 1D and 2D datasets. Next, the chapter elaborates on the DCNN model, including its architecture and simulation results on 1D and 2D datasets. Subsequently, the focus shifts to the DCNN-LSTM model, with explanations of its implementation and simulation outcomes for 1D and 2D datasets. After that, multi-Kernel approach is integrated with hybrid DCNN-LSTM models to further improve the diagnostic accuracy of health conditions of vehicle engine. The discussion then moves to a comparative analysis of AI models, examining the performance of 1D versus 2D datasets and comparing the results of DCNN, DCNN-LSTM, and multi-Kernel-DCNN-LSTM models. Finally, the chapter concludes with an evaluation of the results, highlighting the advantages of AI-based methods for diagnosing misfires, addressing challenges and limitations, and identifying potential areas for future research.

Some results concluded by the AI methods for fault diagnosis are published in the following research papers: *A Novel Approach to Vehicle Engine Misfire Detection with ADXL1002 Accelerometer using EMD Based Image Processing and DCNN-LSTM Model*, published in *Vibration in Physical Systems* and *Comparison of ANN and CNN models for misfire detection in vehicle engine at different RPMs with low-cost ADXL1002 accelerometer*, presented and published in the conference named *2024*

Signal Processing: Algorithms, Architectures, Arrangements, and Applications and Advanced Fault Diagnosis in Rotating Machines using 2D Grayscale Images with Improved Deep Convolutional Neural Networks, presented and published in the conference named *2023 Signal Processing: Algorithms, Architectures, Arrangements, and Applications*.

5.2 Overview of Artificial Intelligence Methods

AI methods have proven to be highly effective for diagnosing vehicle engine misfires by analyzing complex vibration signals. These methods utilize advanced algorithms to identify patterns, anomalies, and features indicative of misfires. Deep learning models, such as DCNN and hybrid architectures like DCNN-LSTM, offer exceptional capabilities in extracting features and detecting faults directly from data. DCNNs are particularly adept at capturing spatial and frequency features from both 1D time-series data and 2D grayscale images of vibration signals, while the LSTM component in hybrid models enhances the ability to learn temporal dependencies and dynamic patterns. By employing these advanced AI approaches with both 1D and 2D data, the accuracy and reliability of vehicle misfire diagnostics are significantly enhanced, offering a robust alternative to traditional diagnostic methods.

While AI methods such as DCNN and hybrid DCNN-LSTM offer significant advancements in fault diagnosis, their application to vehicle misfire detection presents several challenges:

- **Noise and Data Quality:** AI models rely heavily on the quality and quantity of training data. Vibration signals from vehicle misfire datasets are often contaminated with noise from various sources, including mechanical vibrations from other components, engine idling, and environmental factors. These noise elements can obscure fault-related features, making it challenging for models to learn meaningful patterns, especially when the dataset lacks sufficient diversity or noise-robust preprocessing.
- **Model Complexity and Overfitting:** Deep learning models like DCNN and hybrid DCNN-LSTM are inherently complex and require large datasets to prevent overfitting. Limited availability of labeled misfire data or imbalanced datasets can lead to models that perform well during training but fail to generalize to unseen data. Proper regularization, data augmentation, and careful tuning of hyperparameters are necessary to address these issues.
- **Interpretability and Explainability:** Despite their accuracy, AI models often operate as black-box systems, making it difficult to interpret their decision-making process. This lack of transparency can be a barrier in critical applications like vehicle diagnostics, where understanding the root cause of misfires is essential for effective troubleshooting. Developing methods to enhance model explainabil-

ity without compromising performance remains a challenge.

- **Computational Demands:** Training and deploying AI models, particularly hybrid architectures, require significant computational resources. Real-time applications, such as in-vehicle diagnostics, demand lightweight and efficient models, which can be challenging to achieve without sacrificing performance. Balancing computational efficiency with diagnostic accuracy is an ongoing area of research.

5.3 Dataset and Preprocessing

This section provides a comprehensive overview of the vibration dataset recorded from the vehicle engine using the ADXL1002 accelerometer. The details of the experimental setup and recording conditions are given in Chapter 3. The preprocessing step includes data cleaning using EMD, segmentation to prepare 1D time-domain dataset, conversion of segmented data into 2D grayscale images, and visualization of segments using t-distributed Stochastic Neighbor Embedding (t-SNE).

5.3.1 Vibration Dataset

As described in detail in Chapter 3, the data was recorded using a MEMS sensor, the ADXL1002 accelerometer, integrated with the BeagleBone Black control system. The ADXL1002 accelerometer was affixed to the vehicle engine using a magnetic contact. This sensor facilitates the analysis of engine vibrations under various operational conditions. A comprehensive study on the calibration of the ADXL1002 accelerometer interfaced with the BeagleBone Black control system is documented in our previous work [90].

The dataset comprises four entries, each representing a distinct engine condition. The first entry corresponds to a healthy engine state. Subsequent samples depict instances of engine misfires occurring at different RPM levels: the second sample captures a misfire event at 1500 RPM, the third at 2500 RPM, and the fourth at an elevated speed of 3000 RPM. Further details regarding the recorded misfire dataset are provided in Table 5.1.

Table 5.1: Description of the Vehicle Engine Vibration Dataset.

Sr. no.	Health Condition	RPM	Frequency	Data Points
1	Healthy	–	–	882400
2	Misfire	1500 RPM	25 Hz	882400
3	Misfire	2500 RPM	41.67 Hz	882400
4	Misfire	3000 RPM	50 Hz	882400

This dataset, characterized by its high granularity, allows for a thorough exami-

nation of the accelerometer’s sensitivity to variations in engine conditions. It is an invaluable resource for identifying the subtle vibrational nuances associated with both healthy and misfiring states at different RPM levels.

5.3.2 Preprocessing

Preprocessing the vibration dataset for misfire diagnosis in vehicle engines involves addressing unwanted signals and external noises inherent in the complex engine structure. The presence of resonance frequencies from various components adds complexity, making misfire condition diagnosis challenging. To tackle this, dataset cleaning is essential. Figure 5.1 depicts the block diagram of the preprocessing steps detailed in this section.



Figure 5.1: Block diagram for preprocessing steps.

In this approach, EMD is a crucial step. EMD is a signal processing technique that breaks down complex vibration signals into simpler components called IMFs [125, 126]. Each IMF captures a different frequency component of the original signal, helping to isolate and analyze the relevant information. Figure 5.2 visually presents the first-IMF associated with each health condition of the vehicle engine. Each vibration signal comprises 882400 sampled data points, recorded over a duration of 10 seconds with a sampling frequency of 88240 Hz. The first IMF, extracted from each vibration signal, is extremely significant. It represents the most dominant and fundamental frequency components of the signal. This step enables the essential features associated with misfire conditions.

The dataset was then segmented using a window length of 4096 data points and a stride of 2000 data points to create overlapping segments. Subsequently, the segmented dataset was randomized to ensure unbiased processing in subsequent stages. Figure 5.3 presents a selection of random signals from the segmented and shuffled dataset, illustrating the diversity and complexity of the recorded vibration patterns.

Subsequently, the dataset was transformed into 2D grayscale images. Appendix D provides a detailed explanation of the entire process of transforming 1D time-domain data into 2D grayscale images. Figure 5.4 displays a random selection of these 2D grayscale images, which were generated through the preprocessing steps for various health conditions of the vehicle engine. Converting the vibration data into visual representations allows the model to effectively learn and identify patterns associated

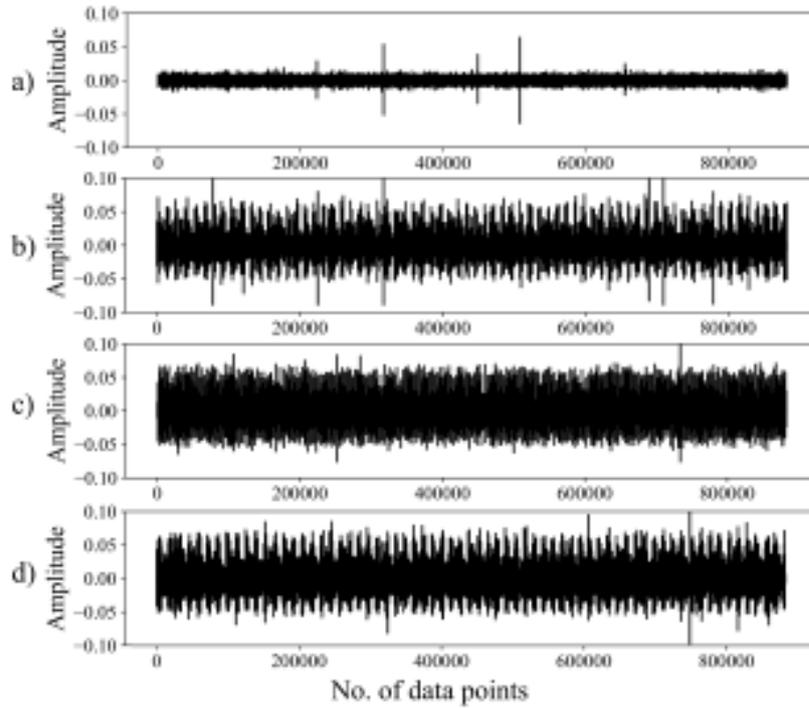


Figure 5.2: First-IMF associated with each health condition of the vehicle engine: a) healthy condition, b) misfire condition at 1500 RPM, c) misfire condition at 2500 RPM, and d) misfire condition at 3000 RPM.

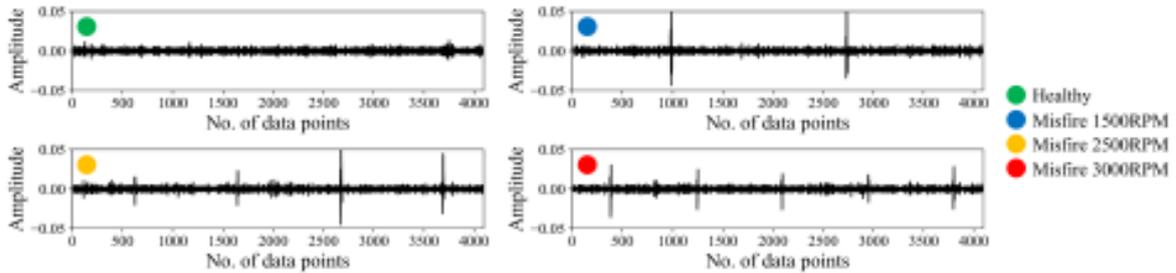


Figure 5.3: Random 1D vibration signals after segmentation and shuffling with a window length of 4096 samples.

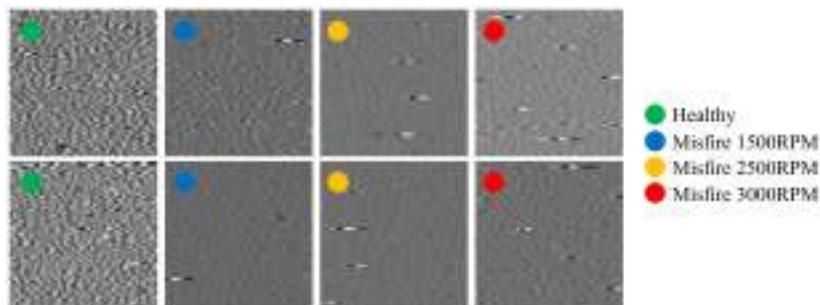


Figure 5.4: Random 2D grayscale images after segmentation and shuffling with a window length of 4096 samples.

with misfire conditions at different RPM levels, thereby enhancing the accuracy and robustness of the diagnostic process.

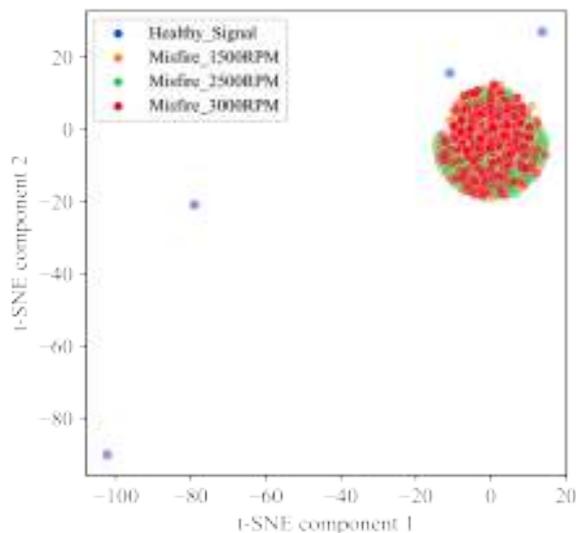


Figure 5.5: t-SNE visualization of vibration data before training.

Before training our dataset using the DCNN or hybrid DCNN-LSTM models, we performed t-SNE to visualize the high-dimensional vibration data obtained from the ADXL1002 accelerometer as illustrated in Figure 5.5. t-SNE is a dimensionality reduction technique that allows us to visualize complex datasets in lower dimensions while preserving local structure, as detailed in Appendix E. In the t-SNE plot, each data point represents a sample from our dataset, and similar samples are grouped together in clusters. This visualization provides insights into the intrinsic structure of the data and can help identify potential patterns or clusters before model training.

While the t-SNE visualization effectively captures the overall structure of the vibration data, it does not distinctly separate the different health conditions of the vehicle engine. Despite the clustering of data points, there is considerable overlap between samples corresponding to different engine conditions. This indicates that the inherent features extracted by t-SNE alone are insufficient for discerning among the various health states of the engine. Therefore, it underscores the need for a more sophisticated model, such as the proposed DCNN and hybrid DCNN-LSTM architectures for 1D and 2D datasets, which can effectively extract and utilize both spatial and temporal features for accurate health diagnostics.

5.4 DCNN Models for Vehicle Diagnostics

To develop a robust misfire detection framework, this research applied and compared multiple deep learning architectures using both 1D and 2D representations of vibration data to diagnose the vehicle engine misfire. In this section, DCNN architectures are presented for 1D and 2D datasets as shown in Figure 5.6.

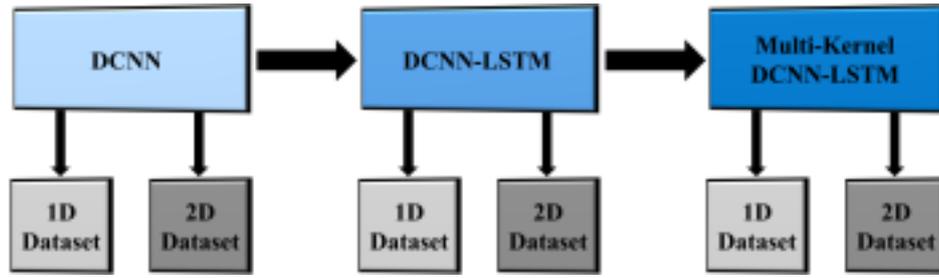


Figure 5.6: Proposed models for misfire diagnosis in vehicle engine.

5.4.1 Implementation of DCNN Models

The DCNN architecture was applied to both 1D and 2D representations of the vibration signals, with the primary structural parameters maintained across both formats. In the 1D configuration, the DCNN model processed the vibration data directly as a 1D time-series signal, capturing essential features through convolutional and pooling layers. This approach leverages DCNN's spatial filtering capabilities to detect temporal features in the 1D signal that may indicate misfire conditions. The 2D configuration, in contrast, utilized transformed vibration signals represented allowing DCNN to exploit spatial features from the grayscale images.

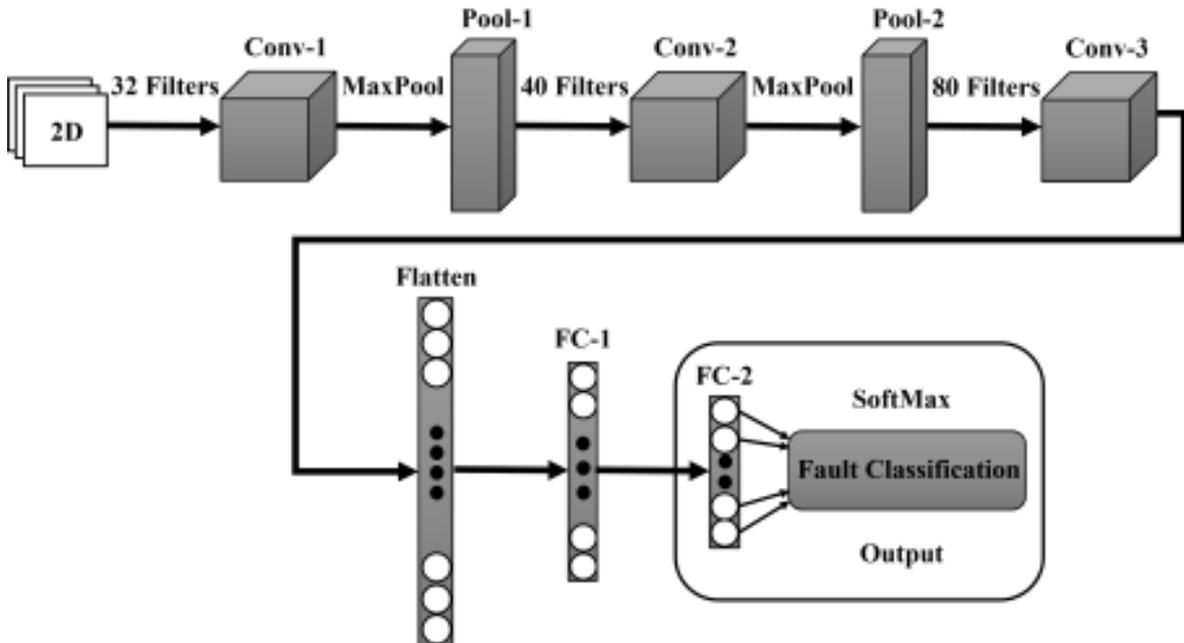


Figure 5.7: Proposed 2D DCNN model for misfire diagnosis in vehicle engine.

Figure 5.7 illustrates the block diagram of the 2D DCNN model used in this chapter. This model is composed of three convolution layers, two MaxPooling layers, one flatten layer, and two fully connected dense layers. The detailed architecture of the 2D DCNN model is given in Table 5.2.

This architecture is composed of several distinct layers, each designed to perform

Table 5.2: 2D DCNN Model.

Layer (Type)	Kernel Size	Output Shape	No. of Parameters
Input Layer	-	(None, 64, 64, 1)	0
Conv2D	6×6	(None, 59, 59, 32)	1184
MaxPooling2D	-	(None, 19, 19, 32)	0
Conv2D	6×6	(None, 14, 14, 40)	46120
MaxPooling2D	-	(None, 7, 7, 40)	0
Conv2D	4×4	(None, 4, 4, 80)	51280
Flatten	-	(None, 1280)	0
Dense	-	(None, 400)	512400
Dense	-	(None, 100)	40100
Dense	-	(None, 4)	404

specific functions in the feature extraction and classification process. The input shape is 64×64 , after that the first layer, Conv2D, has an output shape of (None, 59, 59, 32) and contains 32 filters with a 6×6 kernel size. This layer performs the initial feature extraction by convolving the input data with the filters, resulting in 1184 trainable parameters. The second layer, MaxPooling2D, then down-samples the spatial dimensions of the feature map to (None, 19, 19, 32) using a 3×3 pool size. This layer helps in reducing the dimensionality of the data while retaining the most significant features.

Following this, the another Conv2D layer reduces the output shape to (None, 14, 14, 40) with 40 filters, also utilizing a 6×6 kernel size. This layer further refines the feature map produced by the previous layer, with a total of 46120 trainable parameters. Fourth layer, MaxPooling2D, further down-samples the spatial dimensions of the feature map to (None, 7, 7, 40) using a 2×2 pool size. Fifth layer, Conv2D, has an output shape of (None, 4, 4, 80), contains 80 filters with a 4×4 kernel size.

Next, the Flatten layer converts the 3D tensor output from the previous layer into a 1D vector with a shape of (None, 1280). This step is crucial for transitioning from the convolution layers to the fully connected layers. The first fully connected (Dense) layer follows with 400 units and second with 100 units. These layers applies non-linear transformations to the input data, enabling the network to learn complex representations.

The final Dense layer consists of 4 units, corresponding to the four distinct health states of the vehicle engine: healthy, misfire at 1500 RPM, 2500 RPM, or 3000 RPM. This layer, with 404 trainable parameters, uses a softmax activation function to output the classification probabilities for each health state. This detailed architecture of the 2D DCNN model showcases its capacity for sophisticated feature extraction and classification.

Algorithm 4: 2D DCNN Model for Vehicle Diagnostics

Input : Preprocessed vibration signal data, Grayscale images constructed from 1D time-domain vibration signals,
Input shape (height, width, 1), Number of classes (4), Batch size (16), Number of epochs (50)

Output: Trained DCNN model and visualizing health conditions

```

1 // Define the DCNN model architecture
2 Layer 1: Convolutional Layer
3   • Input shape: (height × width × 1)
4   • Filters: 32; Kernel size: (6 × 6); Activation: ReLU
5 Layer 2: Max-Pooling Layer
6   • Pool size: (3 × 3)
7 Layer 3: Convolutional Layer
8   • Filters: 40; Kernel size: (6 × 6); Activation: ReLU
9 Layer 4: Max-Pooling Layer
10  • Pool size: (2 × 2)
11 Layer 5: Convolutional Layer
12  • Filters: 80; Kernel size: (4 × 4); Activation: ReLU
13 Layer 6: Flatten Layer
14  Convert 2D feature maps into a 1D feature vector
15 Layer 7: Fully Connected Dense Layer
16  • Neurons: 400; Activation: ReLU
17 Layer 8: Fully Connected Dense Layer
18  • Neurons: 100; Activation: ReLU
19 Layer 9: Output Dense Layer
20  • Output Neurons: 4
21  • Activation: Softmax (for classification into 4 health conditions)
22 // Compile the model
23  • Loss function: categorical crossentropy
24  • Optimizer: Adam; Metric: Accuracy
25 // Train the model
26  • Train using the training dataset
27  • Batch size: 16; Number of epochs: 50
28  • Validate using validation dataset
29 // Evaluate the model and visualize results
30 Evaluate model on test dataset
31 Compute and plot confusion matrix for:
32   - Healthy condition
33   - Misfire at 1500 rpm
34   - Misfire at 2500 rpm
35   - Misfire at 3000 rpm

```

The pseudocode for 2D DCNN is given in the Algorithm 4. This allows for a comprehensive analysis of each model’s capability in identifying subtle fault-related patterns within both data formats.

By using the same model structure for both 1D and 2D data, this approach enables a direct comparison of the DCNN’s performance across different data formats, revealing insights into the strengths and limitations of each input representation. For 1D DCNN model, the corresponding model format was updated using 1D convolution and MaxPooling layers.

5.4.2 Simulation Results for 1D and 2D Datasets

The diagnostic performance of the DCNN model was evaluated on both 1D and 2D representations of vibration data. By keeping model parameters consistent across both data formats, this analysis provides insights into the effectiveness of each model architecture in capturing misfire-related features from different representations of vibration signals. The results were evaluated using key metrics, including accuracy, sensitivity, and specificity.

The proposed methodology was implemented using Python 3.8.5 and Anaconda3 on a Windows 10 operating system, running on an Intel(R) Core(TM) i5-8250U CPU 1.60GHz 1.80 GHz, x64-based processor. A detailed exposition of the results is provided in the subsequent content.

After segmentation and shuffling, the dataset was divided into training data and validation data of 70% and 30%, respectively. The DCNN model was trained and validated using the 10 independent runs to verify the consistency of the model, with each run of 50 epochs. These training accuracy, validation accuracy, training loss, and validation loss for 10 independent runs with 50 epoch each are shown in Appendix F.

Figure 5.8 represents the training and validation accuracies for 1D DCNN model, whereas Figure 5.9 shows the training and validation accuracies for 2D DCNN model. Furthermore, Table 5.3 presents a summary of the best model accuracy (maximum accuracy), worst model accuracy (minimum accuracy), average accuracy across all 10 independent runs, and the corresponding standard deviation (SD) for both 1D and 2D datasets using the multi-Kernel-DCNN-LSTM model. This statistical analysis provides insights into the model’s performance consistency, reliability, and variability across

Table 5.3: Evaluation of Maximum, Minimum, Average Accuracy, and Standard Deviation Across 10 Runs for DCNN models.

Dataset	Max. Accuracy(%)	Min. Accuracy(%)	Ave. Accuracy(%)	SD(%)
1D	92.99	90.53	92.01	0.79
2D	97.92	97.35	97.61	0.17

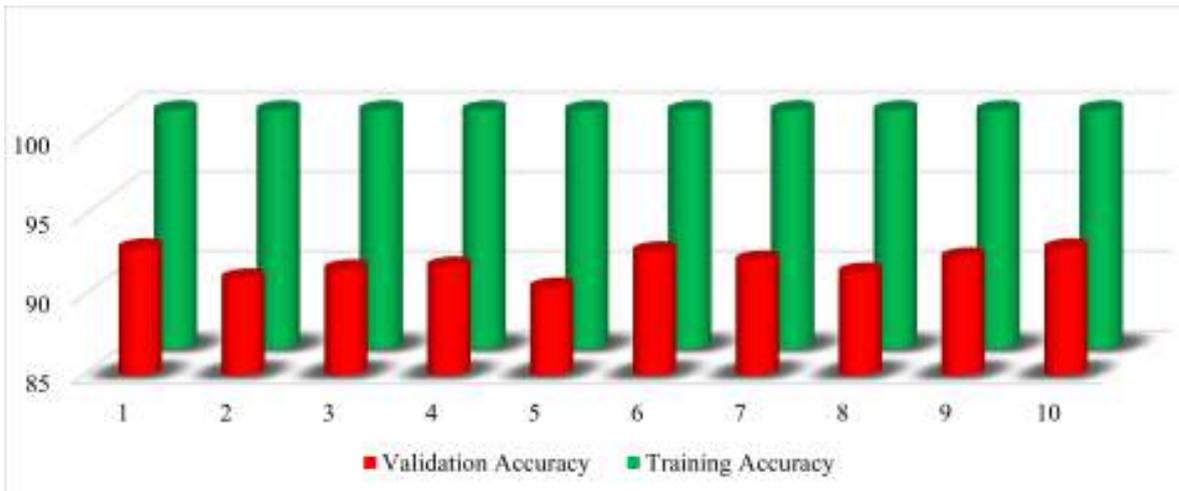


Figure 5.8: Training and validation accuracies over 10 independent runs of 1D DCNN model.

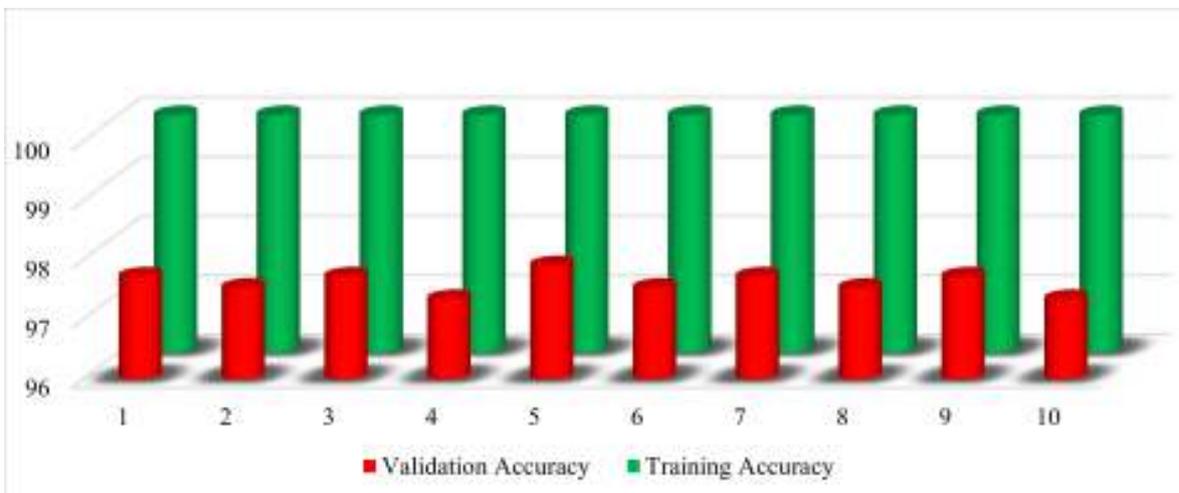


Figure 5.9: Training and validation accuracies over 10 independent runs of 2D DCNN model.

different data representations.

Table 5.3 shows that the average accuracy for 1D dataset is 92.01% whereas for 2D dataset, it is 97.61% with SD of 0.79% and 0.17% for 1D and 2D datasets, respectively. In addition to that, Figure 5.10a displays the training and validation accuracies of the best 1D DCNN model, showing a training accuracy of 100% and a validation accuracy of 92.99%. Similarly, Figure 5.10b illustrates the training and validation accuracies of the best 2D DCNN model, the training accuracy is 100% and validation accuracy achieve 97.92%.

Additionally, Figure 5.11a illustrates the confusion matrix for the 1D DCNN models, indicating a robust performance. Healthy signals are predicted with 98% accuracy, while misfires at 1500 RPM, 2500 RPM, and 3000 RPM are predicted with accuracies of 82%, 98%, and 95%, respectively. Moreover, the confusion matrix in Figure 5.11b

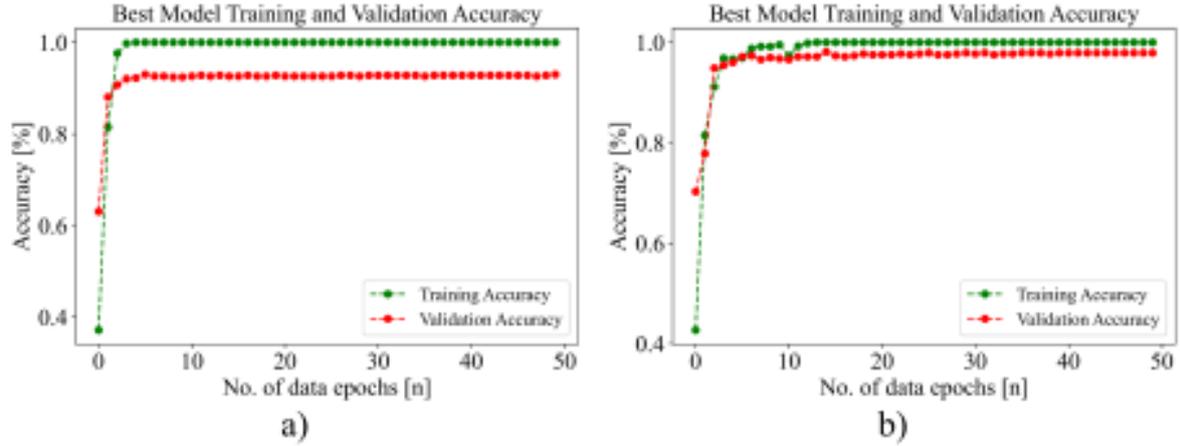


Figure 5.10: Best training accuracy and validation accuracy for: a) 1D DCNN model, b) 2D DCNN model.

Table 5.4: Performance Results for 1D and 2D DCNN Models.

Models	Health Condition	Accuracy	Sensitivity	Specificity	BA	GM
1D DCNN	Healthy Signal	98.39	98.46	98.37	98.41	98.41
	Misfire: 1500RPM	93.32	81.82	97.16	89.49	89.16
	Misfire: 2500RPM	98.69	97.56	99.07	98.31	98.31
	Misfire: 3000RPM	96.24	95.45	96.50	95.98	95.98
2D DCNN	Healthy Signal	99.40	98.37	99.74	99.06	99.06
	Misfire: 1500RPM	97.87	96.15	98.44	97.30	97.29
	Misfire: 2500RPM	98.47	96.95	98.97	97.96	97.96
	Misfire: 3000RPM	100	100	100	100	100

outlines the performance of the 2D DCNN model. Healthy signals are predicted with 98% accuracy, while misfires at 1500 RPM, 2500 RPM, and 3000 RPM are predicted with accuracies of 96%, 97%, and 100%, respectively. These results highlight the varying performance of the different models in accurately diagnosing engine misfires across different RPM levels. These results showcase the effectiveness of our model in accurately classifying different engine conditions based on vibration data.

Following the accuracy assessments, Figure 5.12a&b provides a t-SNE representation of the feature embeddings generated by the 2D DCNN, and 1D DCNN models. This visualization offers insights into the distribution and clustering of the extracted features within the high-dimensional space, as detailed in Appendix E. By examining the proximity of data points in the t-SNE plot, we can discern patterns and relationships between different engine conditions, further elucidating the models' discriminative capabilities in capturing subtle variations associated with healthy and misfiring states across various RPM levels.

Moreover, in Table 5.4 a detailed analysis of the conclusive performance results attained by DCNN model is presented. The evaluation involved implementing the pro-

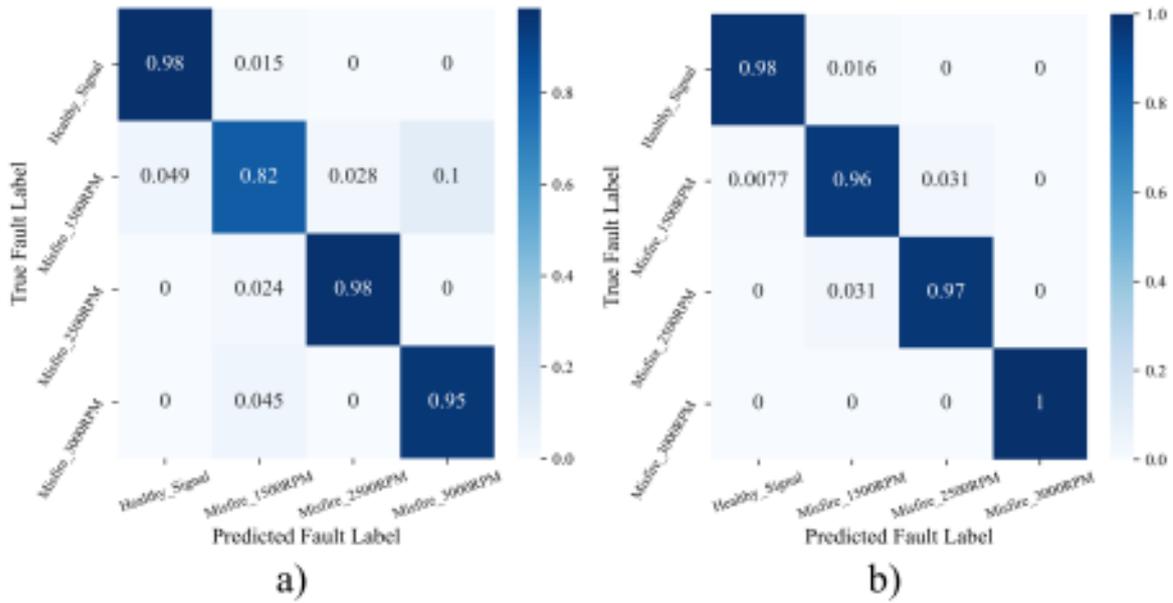


Figure 5.11: Confusion matrices representing true fault versus predicted fault for engine misfire at different RPMs: a) 1D DCNN model, and b) 2D DCNN model.

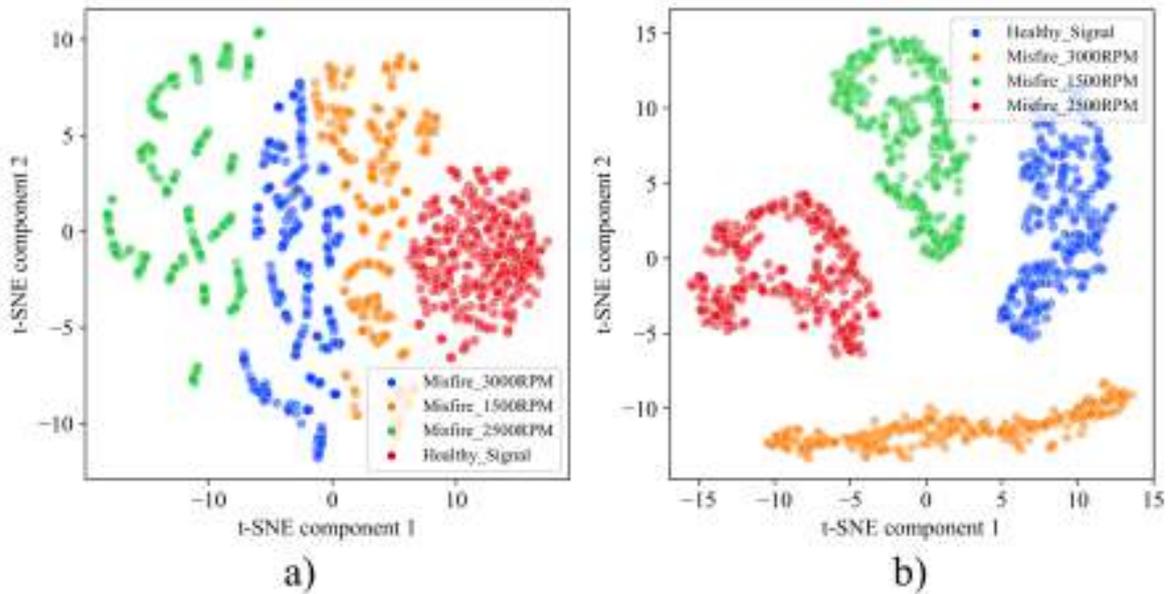


Figure 5.12: t-SNE visualization of vibration data after training; a) 1D DCNN model, and b) 2D DCNN model.

posed 1D and 2D DCNN architectures on test sets specifically tailored for the misfire dataset, as expounded in Table 5.1. To assess the system's efficacy comprehensively, we considered multiple performance indicators. First, the accuracy metric gauges the overall correctness of our system's predictions, indicating the proportion of accurately identified instances. Sensitivity, or true positive rate, measures the system's adeptness at correctly identifying positive instances among all actual positive instances. Specificity assesses the system's precision in identifying negative instances among all

actual negative instances. The Balanced Accuracy (BA) metric provides a nuanced evaluation, considering both sensitivity and specificity to ensure a balanced assessment across different classes or outcomes. Additionally, the Geometric Mean (GM), derived as the square root of the product of sensitivity and specificity, offers a holistic measure of overall classification performance, particularly in scenarios with class imbalances. These metrics were computed through specific formulas or algorithms given in the following equations, showcasing the system's proficiency across various dimensions of classification accuracy and effectiveness. Following equations are used to compute the performance results.

$$Accuracy (\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (5.1)$$

$$Sensitivity (\%) = \frac{TP}{TP + FN} \times 100 \quad (5.2)$$

$$Specificity (\%) = \frac{TN}{TN + FP} \times 100 \quad (5.3)$$

$$BA (\%) = \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) \times 100 \quad (5.4)$$

$$GM (\%) = \sqrt{\frac{TP}{TP + FN} \times \frac{TN}{TN + FP}} \times 100 \quad (5.5)$$

To determine true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) for a dataset featuring classes such as healthy signals, misfire at 1500RPM, misfire at 2500RPM, and misfire at 3000RPM, a comparison must be made between the model's predictions and the actual class labels. True positives occur when the model accurately predicts positive cases for each class, like correctly identifying misfire at 1500RPM. Conversely, false positives arise when the model incorrectly predicts positive cases, such as misidentifying a healthy signal as misfire at 1500RPM. True negatives represent instances where the model correctly predicts negative cases, like identifying a healthy signal as not a misfire at 1500RPM. On the other hand, false negatives occur when the model incorrectly predicts negative cases, such as misidentifying misfire at 1500RPM as a healthy signal. These numerical values are subsequently employed in calculating metrics like precision, recall, and accuracy, contributing to a more comprehensive evaluation of the model's performance for each class.

Figure 5.13 presents a bar graph illustrating the accuracies reported in Table 5.4 for the 1D DCNN model, offering a visual representation of its performance in diagnosing engine misfires. Similarly, Figure 5.14 depicts a bar graph showcasing the accuracies from Table 5.4 for the 2D DCNN model. These bar graphs serve as visual aids for

comparing the performance of each model variant, providing a concise overview of their respective accuracies in detecting engine misfires across different RPM levels. Through these graphical representations, researchers and practitioners can easily assess and compare the diagnostic capabilities of each model, facilitating informed decision-making in the implementation of engine health monitoring systems.

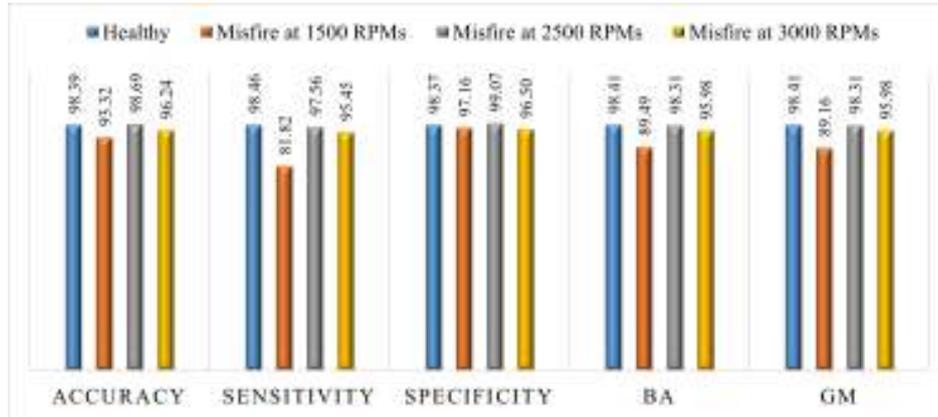


Figure 5.13: Performance results of 1D DCNN model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.

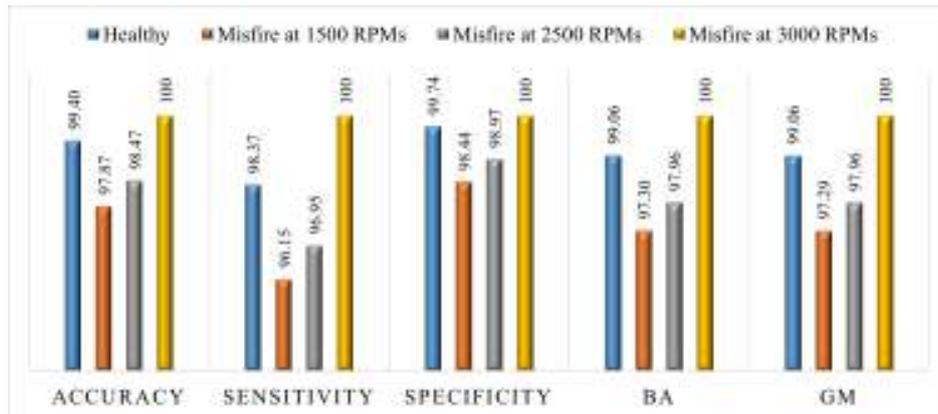


Figure 5.14: Performance results of 2D DCNN model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.

5.5 DCNN-LSTM Models for Vehicle Diagnostics

To enhance the accuracy of misfire detection, this research extended the deep learning framework by integrating LSTM layers into the DCNN architectures. This hybrid DCNN-LSTM model leverages both spatial and temporal features of the vibration data to diagnose vehicle engine misfires more effectively. The detailed architecture of the DCNN-LSTM model is presented in this section.

5.5.1 Implementation of DCNN-LSTM Models

To further enhance fault diagnosis, DCNN-LSTM models were applied to both 1D and 2D representations of vibration signals. The DCNN component allows for deeper feature extraction from the input data, capturing complex diagnostic features that might otherwise be missed in shallow architectures. By incorporating LSTM layers, the DCNN-LSTM model integrates temporal modeling capabilities, making it well-suited to detect misfire events that exhibit sequential patterns over time.

The structure of the DCNN-LSTM model for 2D inputs is outlined in Figure 5.15. This model architecture begins with convolution layers that capture critical spatial and temporal patterns in the data, followed by LSTM layers that identify long-term dependencies associated with engine misfire.

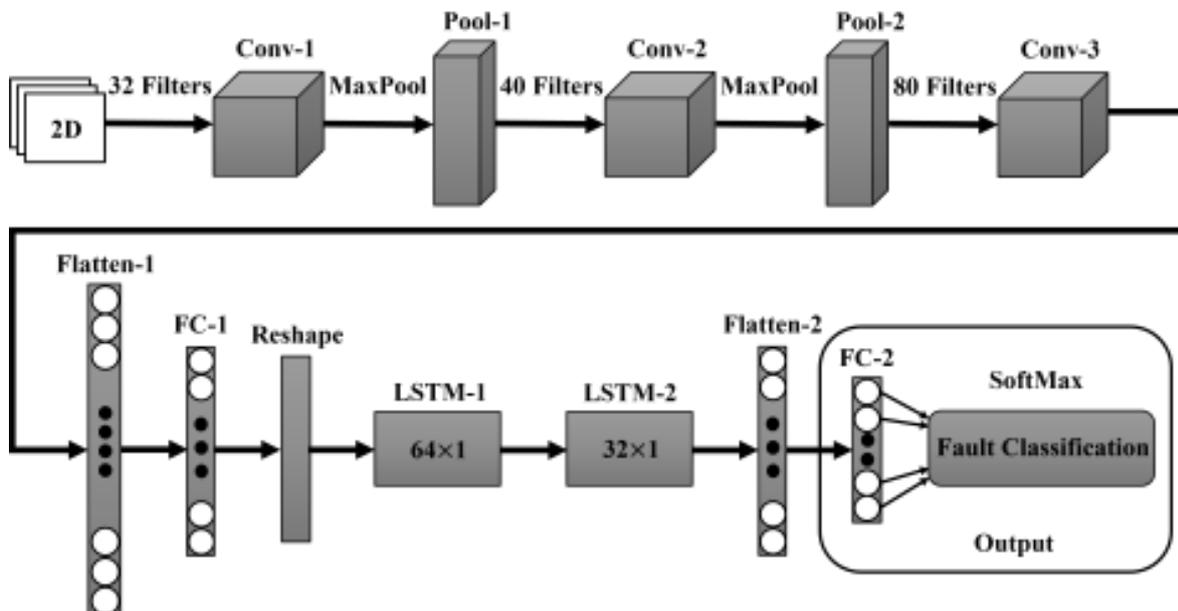


Figure 5.15: Proposed 2D DCNN-LSTM model for misfire diagnosis in vehicle engine.

The DCNN-LSTM model designed for health diagnostics of vehicle engines, leveraging data from the low-cost ADXL1002 accelerometer, comprises a sequence of distinct layers tailored for comprehensive feature extraction and temporal modeling. In the hybrid DCNN-LSTM model, the convolutional layers in the DCNN architecture preserve the input and output dimensions by employing `padding='same'`. The kernel sizes and other parameters remain consistent with those used in the DCNN model implemented in the previous section. Following the convolutional layers, a Flatten layer reshapes the feature maps into a one-dimensional vector, enabling a smooth transition to the Dense layer, which applies non-linear transformations with 128 units. Subsequently, a Reshape layer prepares the data for LSTM processing by converting the output into an appropriate three-dimensional tensor format.

Integration of LSTM layers is a distinctive feature of the proposed model. The

input is reshaped for compatibility with the sequential nature of LSTM, followed by two LSTM layers with 64 and 32 units, respectively. These layers enable the model to capture temporal dependencies and long-term patterns within the vibration signals. LSTM Layer with 64 units captures temporal dependencies, complemented by LSTM Layer with 32 units for enhanced temporal modeling. Flatten Layer condenses the output for final processing in Dense Layer, producing a (None, 4) output shape representing the distinct health states of the vehicle engine. This final dense layer, activated by softmax, outputs probabilities across the specified number of classes (number classes = 4), representing different engine conditions (1500 RPM, 2500 RPM, 3000 RPM, and a healthy signal). This meticulously crafted architecture synergizes convolutional and LSTM layers to yield accurate health diagnostics, crucial for maintenance and performance optimization. The model is compiled using categorical cross-entropy as the loss function, Adam optimizer for parameter optimization, and accuracy as the evaluation metric. Table (5.5) illustrates the overall proposed model described above.

Table 5.5: 2D DCNN-LSTM Model.

Layer (Type)	Kernel Size	Output Shape	No. of Parameters
Input Layer	-	(None, 64, 64, 1)	0
Conv2D	6×6	(None, 64, 64, 32)	1184
MaxPooling2D	-	(None, 21, 21, 32)	0
Conv2D	6×6	(None, 21, 21, 40)	46120
MaxPooling2D	-	(None, 10, 10, 40)	0
Conv2D	4×4	(None, 10, 10, 80)	51280
Flatten	-	(None, 8000)	0
Dense	-	(None, 128)	1024128
Reshape	-	(None, 1, 128)	0
LSTM	-	(None, 1, 64)	49408
LSTM	-	(None, 32)	12416
Flatten	-	(None, 32)	0
Dense	-	(None, 4)	132

The proposed 2D DCNN-LSTM model for health diagnostics of vehicle engines is depicted in Figure 5.15. The architecture comprises a sequence of distinct layers, each meticulously tailored to process and extract meaningful insights from the vibration data acquired through the low-cost ADXL1002 accelerometer. This comprehensive 2D DCNN-LSTM architecture enables our model to effectively learn from the vibration data and make accurate predictions regarding misfire conditions across different engine speeds, ultimately enhancing the diagnostic capabilities in vehicle engine health monitoring. The pseudocode for 2D DCNN-LSTM model is given in the Algorithm 5.

Algorithm 5: 2D DCNN-LSTM Model for Vehicle Diagnostics

Input : Preprocessed vibration signal data, Grayscale images constructed from 1D time-domain vibration signals, input shape $(height, width, 1)$, number of classes (4), batch size (16), number of epochs (50)

Output: Trained DCNN-LSTM layers, and visualizing health conditions

```

1 // Add DCNN layers
2 Add a 2D convolutional layer with:
3   • Input shape:  $(height, width, 1)$ 
4   • Filters: 32; Kernel size:  $(6 \times 6)$ ; Activation: ReLU;
5 Add a max-pooling layer with: Pool size:  $(3 \times 3)$ ;
6 Add another 2D convolutional layer with:
7   • Filters: 40; Kernel size:  $(6 \times 6)$ ; Activation: ReLU;
8 Add a max-pooling layer with: Pool size:  $(2 \times 2)$ ;
9 Add another 2D convolutional layer with:
10  • Filters: 80; Kernel size:  $(4 \times 4)$ ; Activation: ReLU;
11 Flatten layer to convert 2D feature maps into a 1D feature vector;
12 Add a fully connected dense layer with: Neurons: 128; Activation: ReLU
13 // Reshape output to fit the LSTM input
14 Reshape the output to shape  $(1 \times 128)$  for LSTM input
15 // Add LSTM layers
16 Add the first LSTM layer with: Units: 64; Return sequences: True
17 Add the second LSTM layer with: Units: 32; Return sequences: False
18 Flatten the output from the LSTM layers
19 Add a fully connected output layer with: Neurons: 4; Activation: Softmax
20 // Compile the model
21 Compile the model using:
22   • Loss function: categorical_crossentropy
23   • Optimizer: Adam; Metric: accuracy
24 // Train the model
25   • Train using the training dataset
26   • Batch size: 16; Number of epochs: 50
27   • Validate using validation dataset
28 // Evaluate the model and visualize results
29 Evaluate model on validation dataset
30 Compute and plot confusion matrix for:
31   - Healthy condition
32   - Misfire at 1500 rpm
33   - Misfire at 2500 rpm
34   - Misfire at 3000 rpm

```

Following this, a 1D DCNN-LSTM model is also developed to serve as a comparative baseline. The 1D DCNN-LSTM model retains the same number of layers as the 2D variant. However, the primary difference lies in the input shape and the nature of the convolutional and max pooling layers, which are adapted to 1D operations. Specifically, the input data for the 1D DCNN-LSTM model is the sequential vibration data, and both the convolutional layers and max pooling layers operate in one dimension. This comparison allows us to evaluate the impact of using 1D versus 2D representations of the vibration data on the model's performance in diagnosing engine misfires.

5.5.2 Simulation Results for 1D and 2D Datasets

This section delineates the findings obtained from the experimental analysis conducted on the vehicle vibration dataset using 1D and 2D DCNN-LSTM models. As described earlier, prior to applying the proposed 1D and 2D DCNN-LSTM models, we performed preprocessing to clean the dataset from undesired signals and external noises stemming from the intricate dynamics of the vehicle engine. Once the vibration data underwent cleaning, it was transformed into 2D grayscale images to serve as input for the proposed 2D DCNN-LSTM model. The 1D DCNN-LSTM model retained the same layer structure as the 2D DCNN-LSTM model but processed the sequential 1D vibration data directly whereas 2D DCNN-LSTM model utilized 2D grayscale images.

The DCNN-LSTM model was trained and validated using 10 independent runs to verify the consistency of the model with each run of 50 epochs. For each run, Figure 5.16 illustrates the training and validation accuracies of 1D DCNN-LSTM model and Figure 5.17 represents the training and validation accuracies of 2D DCNN-LSTM model. Among the 10 independent runs, the best model training accuracy and validation accuracy for 1D DCNN-LSTM model is displayed for 50 epochs in Figure 5.18a;

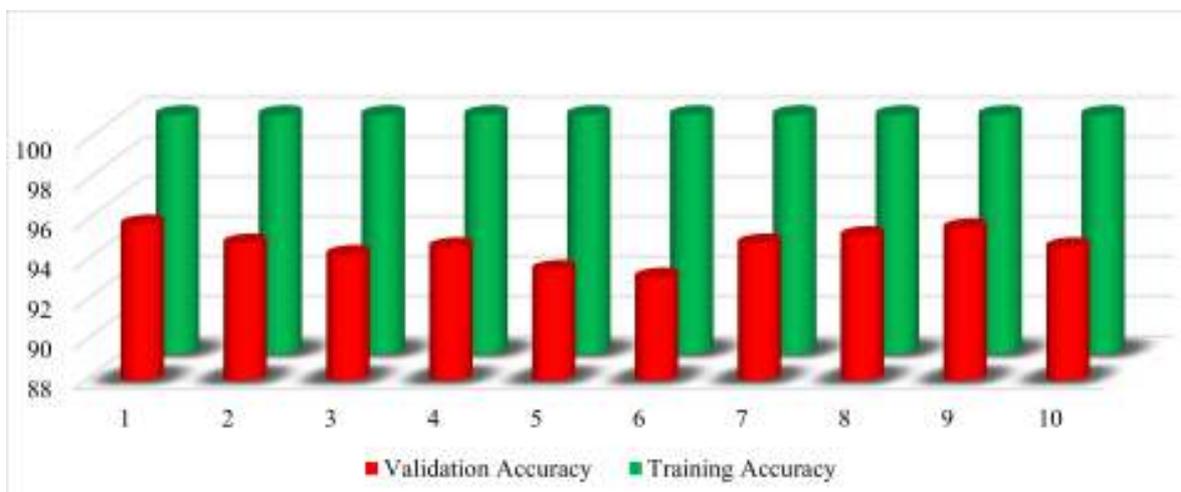


Figure 5.16: Training and validation accuracies over 10 independent runs of 1D DCNN-LSTM model.

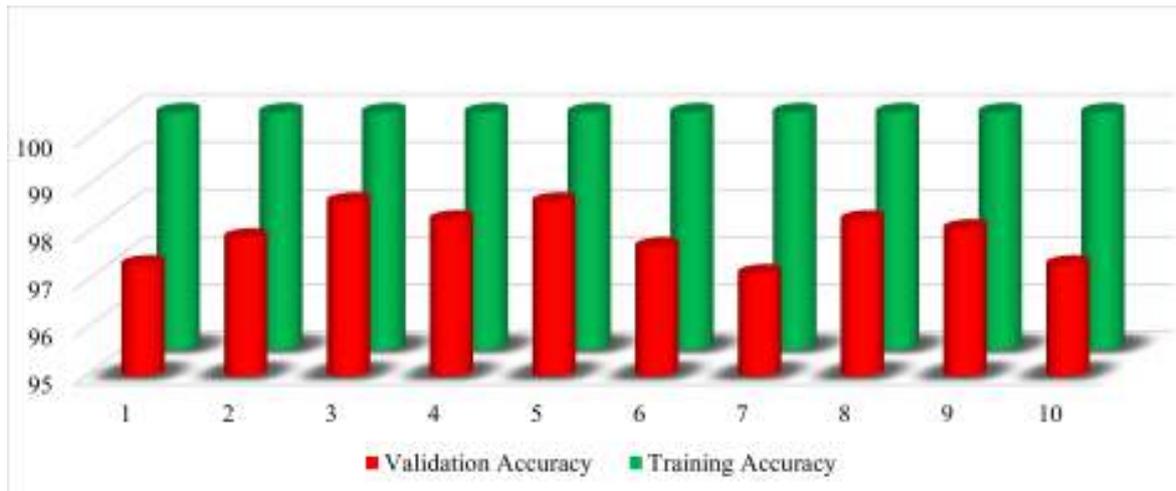


Figure 5.17: Training and validation accuracies over 10 independent runs of 2D DCNN-LSTM model.

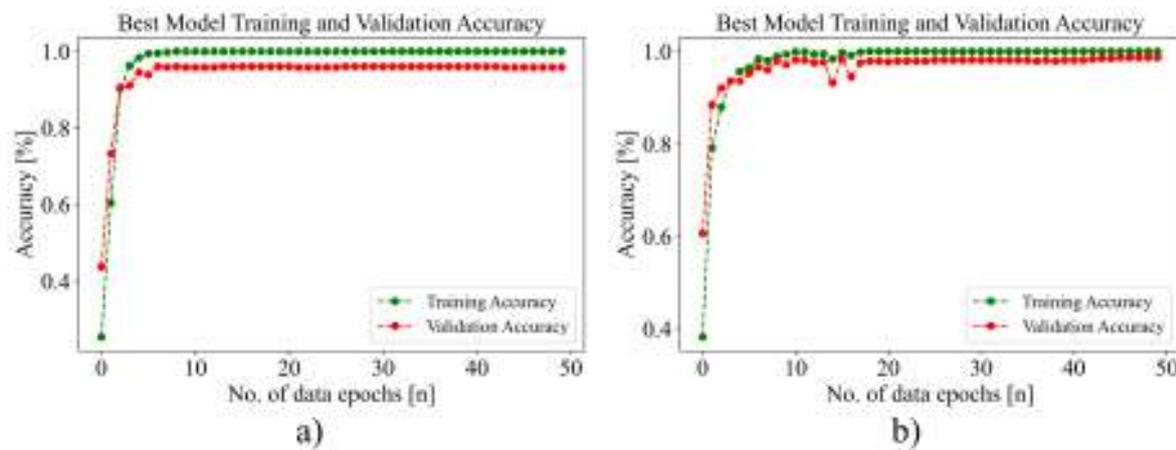


Figure 5.18: Best training accuracy and validation accuracy for: a) 1D DCNN-LSTM model, b) 2D DCNN-LSTM model.

Table 5.6: Evaluation of Maximum, Minimum, Average Accuracy, and Standard Deviation Across 10 Runs for DCNN-LSTM models.

Dataset	Max. Accuracy	Min. Accuracy	Ave. Accuracy	SD
1D	95.83	93.18	94.70	0.79
2D	98.67	97.16	97.95	0.52

showcasing a training accuracy of 100% and a commendable validation accuracy of 95.83%. Similarly, Figure 5.18b depicts the training accuracy and validation accuracy of the best 2D DCNN-LSTM model among 10 independent runs, where the training accuracy also achieves 100%, with the validation accuracy at 98.67%.

The average validation accuracy of 10 independent runs for 1D and 2D DCNN-LSTM models are 94.70% and 97.95%, respectively. In addition to that average accuracy, the minimum accuracy for worst 1D DCNN-LSTM model is 93.18% and for

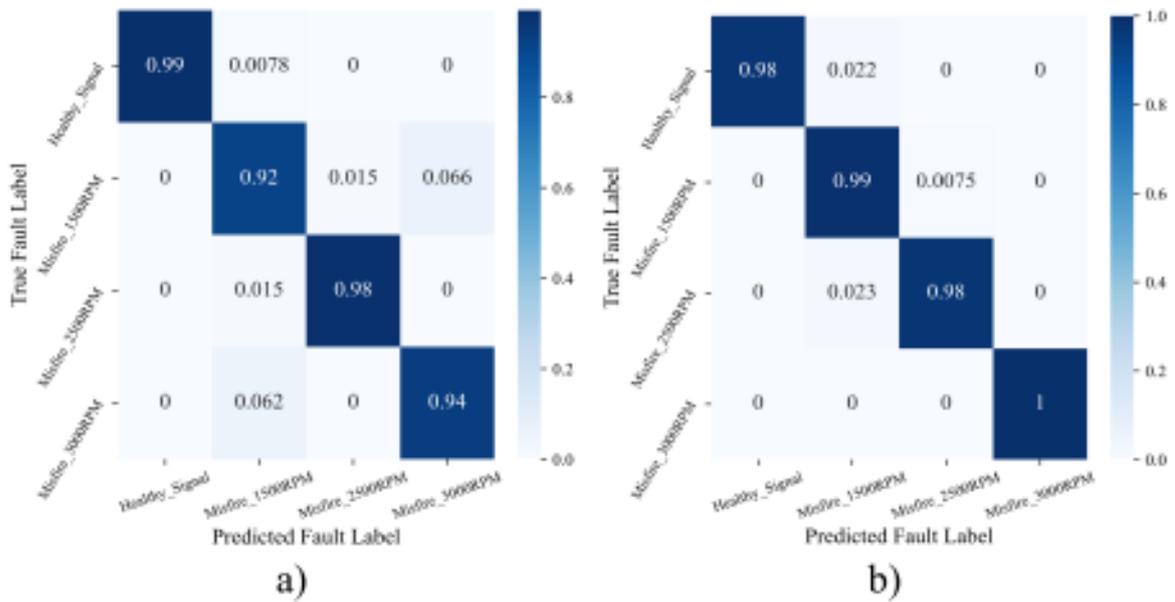


Figure 5.19: Confusion matrices representing true fault versus predicted fault for engine misfire at different RPMs: a) 1D DCNN-LSTM model, b) 2D DCNN-LSTM model.

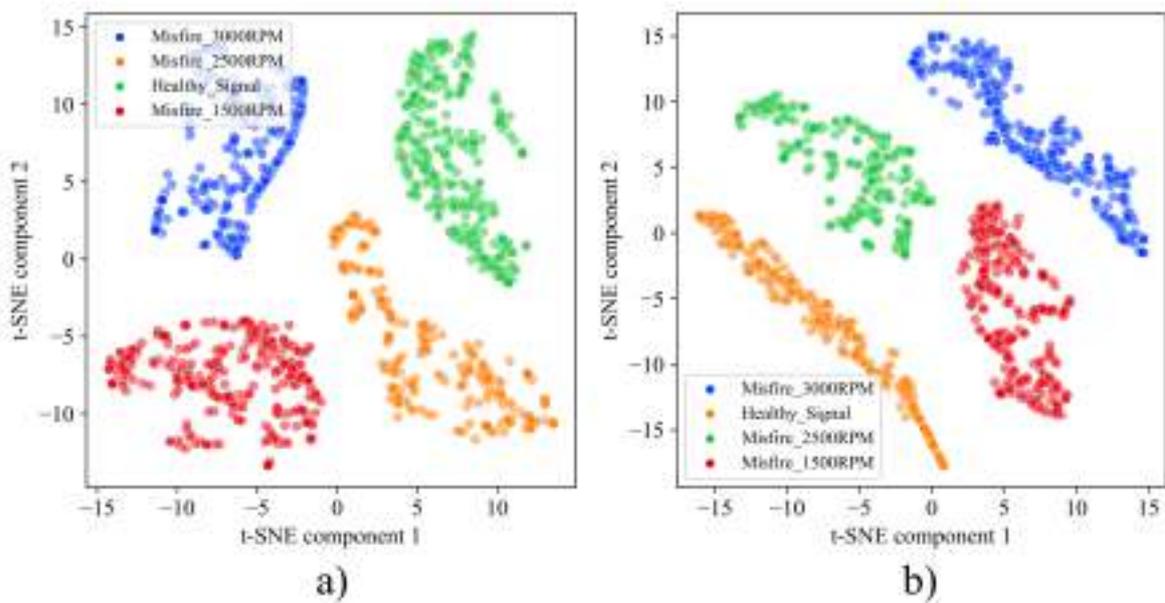


Figure 5.20: t-SNE visualization of vibration data after training; a) 1D DCNN-LSTM model, b) 2D DCNN-LSTM model.

2D DCNN-LSTM model, the accuracy is 97.16%. Table 5.6 shows the best accuracy, worst accuracy, average accuracy, and SD for both 1D and 2D DCNN-LSTM models.

Additionally, Figure 5.19 presents the confusion matrices for the best DCNN-LSTM models, providing a visual representation of the accuracy for each class. According to the confusion matrices depicted in Figure 5.19a, the proposed 1D DCNN-LSTM model demonstrates high prediction accuracy across different engine conditions. Specifically, healthy signals are predicted with 99% accuracy, while misfires occurring at 1500 RPM,

2500 RPM, and 3000 RPM are predicted with 92%, 98%, and 94% accuracy, respectively. Similarly, the confusion matrix presented in Figure 5.19b outlines the performance of the best 2D DCNN-LSTM model. Healthy signals are predicted with 98% accuracy, while misfires at 1500 RPM, 2500 RPM, and 3000 RPM are predicted with accuracies of 99%, 98%, and 100%, respectively.

Following the accuracy assessments, Figure 5.20 provides a t-SNE representation of the feature embeddings generated by the 1D DCNN-LSTM and 2D DCNN-LSTM models. This visualization offers insights into the distribution and clustering of the extracted features within the high-dimensional space, as detailed in Appendix E. By examining the proximity of data points in the t-SNE plot, we can discern patterns and relationships between different engine conditions, further elucidating the models' discriminative capabilities in capturing subtle variations associated with healthy and misfiring states across various RPM levels.

Table 5.7: Performance Results for 1D and 2D DCNN-LSTM Models.

Models	Health Condition	Accuracy	Sensitivity	Specificity	BA	GM
1D DCNN-LSTM	Healthy Signal	99.80	99.22	100	99.61	99.61
	Misfire: 1500RPM	95.88	91.97	97.19	94.58	94.54
	Misfire: 2500RPM	99.26	98.50	99.51	99.00	99.00
	Misfire: 3000RPM	96.82	93.85	97.81	95.83	95.81
2D DCNN-LSTM	Healthy Signal	99.45	97.81	100	98.91	98.90
	Misfire: 1500RPM	98.70	99.25	98.51	98.88	98.88
	Misfire: 2500RPM	99.24	97.73	99.75	98.74	98.73
	Misfire: 3000RPM	100	100	100	100	100

Table 5.7 presents the performance evaluation of 1D and 2D DCNN-LSTM models across different health conditions, including healthy signals and misfire events at various engine speeds. Overall, both models exhibit high classification accuracy, sensitivity, specificity, BA, and GM, demonstrating their effectiveness in diagnosing vehicle misfire conditions. The 2D DCNN-LSTM model outperforms the 1D variant in most cases, particularly in misfire detection at 1500 RPM and 3000 RPM, where it achieves superior sensitivity and specificity. Notably, the 2D DCNN-LSTM model attains a perfect score of 100% across all metrics for the misfire condition at 3000 RPM, indicating its robustness in handling higher-speed misfire detection. While the 1D model also performs well, with accuracy exceeding 95% for all cases, its sensitivity is relatively lower for misfire conditions at 1500 RPM and 3000 RPM compared to the 2D model. These results suggest that leveraging two-dimensional representations enhances feature extraction capabilities, leading to improved fault diagnosis performance, particularly in challenging conditions. This flawless performance underscores the model's ability to effectively identify engine abnormalities across a wide range of operational conditions,

providing automotive engineers and technicians with a reliable tool for engine health monitoring and maintenance.

The performance across different classes reaffirms the model's suitability for real-world applications, where accurate and robust diagnostic capabilities are paramount for ensuring vehicle safety and performance. Moreover, the results highlight the effectiveness of incorporating both spatial and temporal information, as demonstrated by the DCNN-LSTM models, particularly the 2D DCNN-LSTM model, which leverages 2D grayscale images of vibration data for enhanced feature extraction and temporal modeling, leading to exceptional diagnostic accuracy in detecting engine misfires.

Figure 5.21 presents a bar graph illustrating the accuracies reported in Table 5.7 for the 1D DCNN model, offering a visual representation of its performance in diagnosing engine misfires. Similarly, Figure 5.22 displays a bar graph presenting the accuracies detailed in Table 5.7 for the 2D DCNN-LSTM model. These bar graphs serve as visual aids for comparing the performance of each model variant, providing a concise overview of their respective accuracies in detecting engine misfires across different RPM levels.

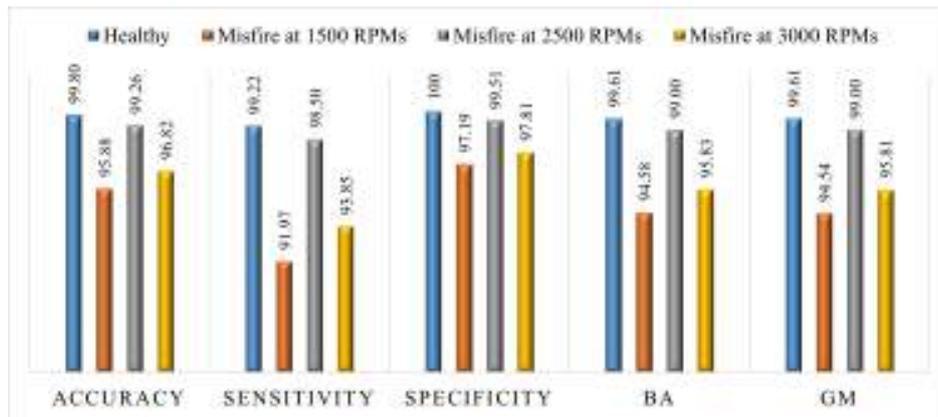


Figure 5.21: Performance results of 1D DCNN-LSTM model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.

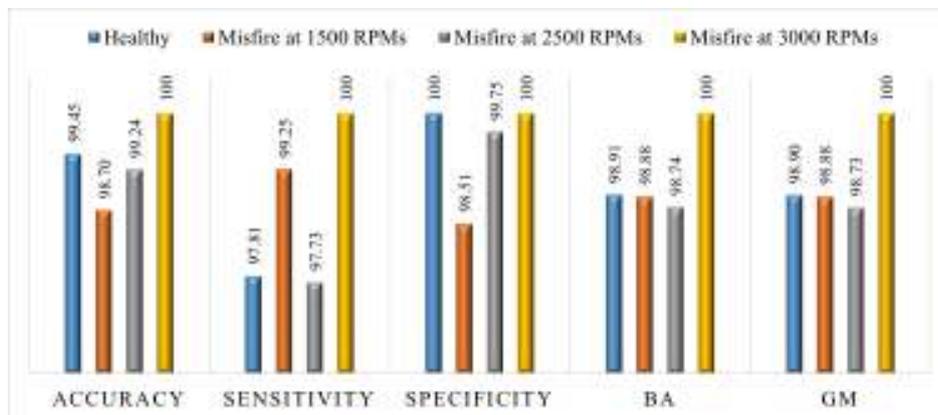


Figure 5.22: Performance results of 2D CNN-LSTM model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.

Through these graphical representations, researchers and practitioners can easily assess and compare the diagnostic capabilities of each model, facilitating informed decision-making in the implementation of engine health monitoring systems.

5.6 multi-Kernel-DCNN-LSTM Models for Vehicle Diagnostic

From the previous section, we have seen that hybrid DCNN-LSTM models for 1D and 2D datasets enhance the diagnostic performance for vehicle engine misfire. In this section, the hybrid models for 1D and 2D datasets are further improved using the multi-Kernel approach. The detailed model implementation and simulation results are given bellow.

5.6.1 Implementation of multi-Kernel-DCNN-LSTM Models

To improve the performance of the hybrid DCNN-LSTM models for 1D and 2D datasets, multi-Kernel approach is implemented in this section. The DCNN layers are divided into two heads as shown in Figure 5.23. This figure illustrates the block diagram

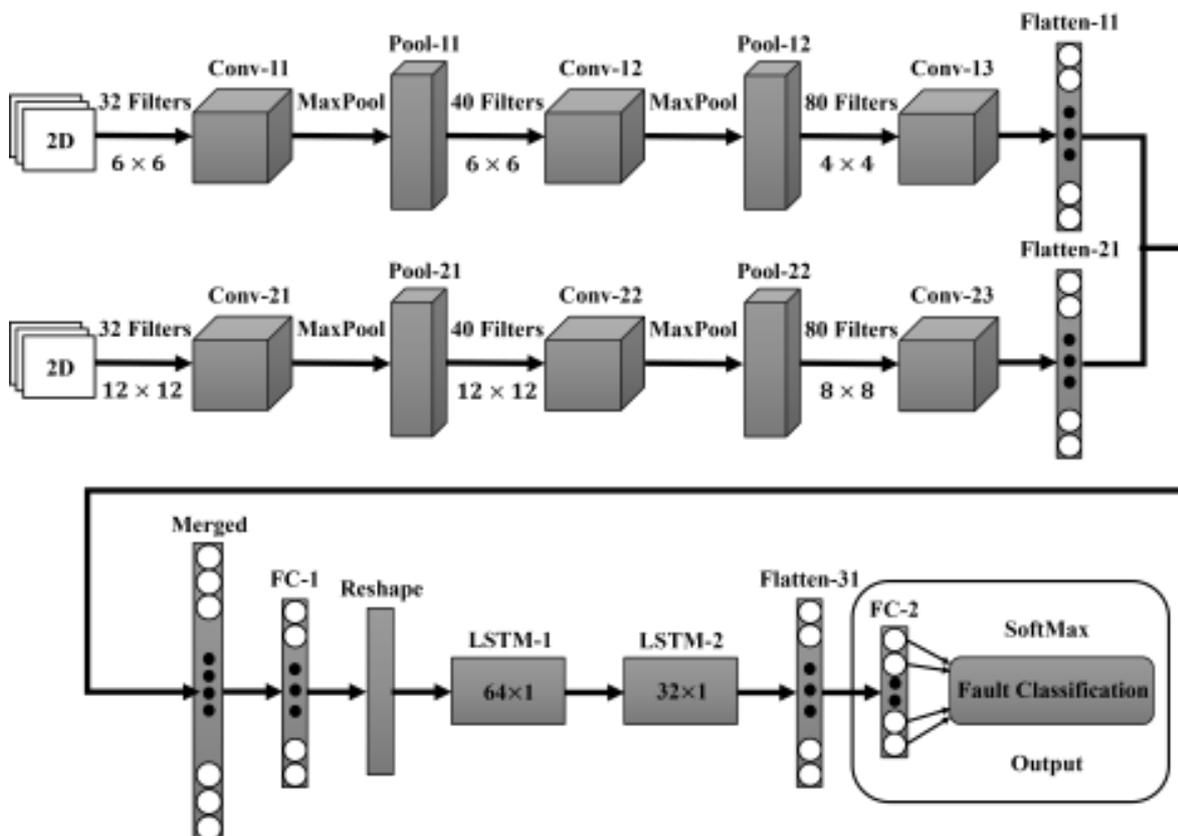


Figure 5.23: Proposed multi-Kernel-2D-DCNN-LSTM model for misfire diagnosis in vehicle engine.

of the proposed multi-Kernel-DCNN-LSTM model. Multi-Kernel-DCNN-LSTM model retains the same architecture as the hybrid DCNN-LSTM model but introduces two parallel DCNN blocks with different kernel sizes. Despite the variation in kernel sizes, both DCNN blocks share identical convolutional and max-pooling layer configurations. The outputs from these DCNN layers are concatenated for further processing. The subsequent LSTM layers in the multi-Kernel-DCNN-LSTM model remain unchanged from those in the hybrid DCNN-LSTM model, ensuring consistency in temporal feature extraction.

5.6.2 Simulation Results for 1D and 2D Datasets

After segmentation and shuffling, the dataset was divided into training data and validation data of 70% and 30%, respectively. The proposed multi-Kernel-DCNN-LSTM

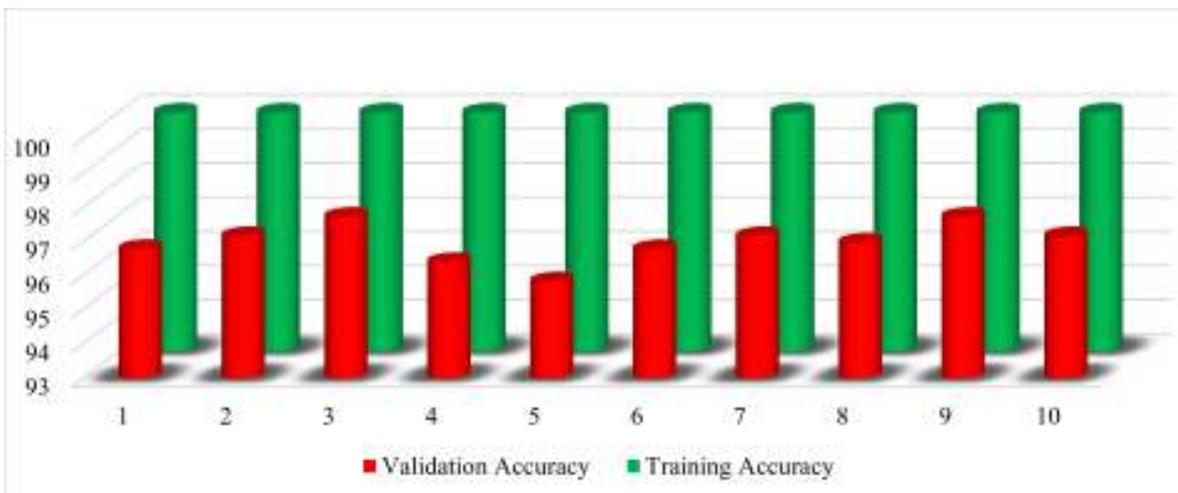


Figure 5.24: Training and validation accuracies over 10 independent runs for 1D dataset.

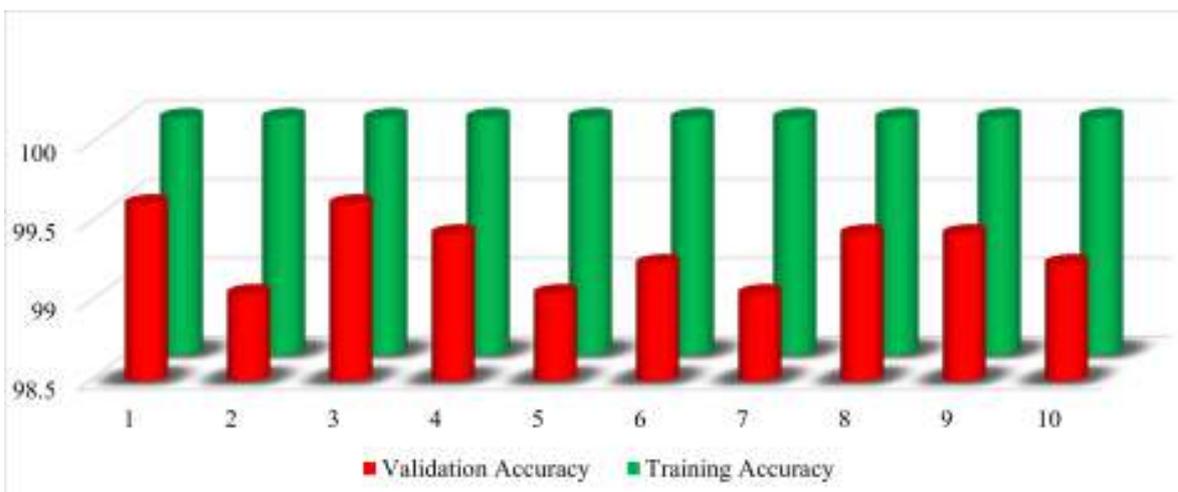


Figure 5.25: Training and validation accuracies over 10 independent runs for 2D dataset.

model was trained for 10 independent runs, with each run consisting of 50 training epochs. The training accuracy, validation accuracy, training loss, and validation loss for 10 independent runs with 50 epoch each are shown in Appendix F. This approach ensures robustness by evaluating the model's performance across multiple training iterations, mitigating the effects of stochastic variations in weight initialization and optimization. Figure 5.24 illustrates the training and validation accuracies for 1D dataset, whereas Figure 5.25 shows the training and validation accuracies for 2D grayscale images. Furthermore, Table 5.8 presents a summary of the best model accuracy (maximum accuracy), worst model accuracy (minimum accuracy), average accuracy across all 10 independent runs, and the corresponding SD for both 1D and 2D datasets using the multi-Kernel-DCNN-LSTM model. This statistical analysis provides insights into the model's performance consistency, reliability, and variability across different data representations.

Table 5.8 shows that the maximum accuracy for 1D dataset is 97.73% whereas for 2D dataset, it is 99.62%. Figure 5.26a displays the training and validation accuracies of the 1D multi-Kernel-DCNN-LSTM model, showing a training accuracy of 100% and a validation accuracy of 97.73%. Similarly, Figure 5.26b illustrates the training and validation accuracies of 2D multi-Kernel-DCNN-LSTM model, where the training accuracy and validation accuracy achieve 100% and 99.62%, respectively.

Table 5.8: Evaluation of Maximum, Minimum, Average Accuracy, and Standard Deviation Across 10 Runs for multi-Kernel DCNN-LSTM models.

Dataset	Max. Accuracy	Min. Accuracy	Ave. Accuracy	SD
1D	97.73	95.83	96.97	0.54
2D	99.62	99.05	99.32	0.21

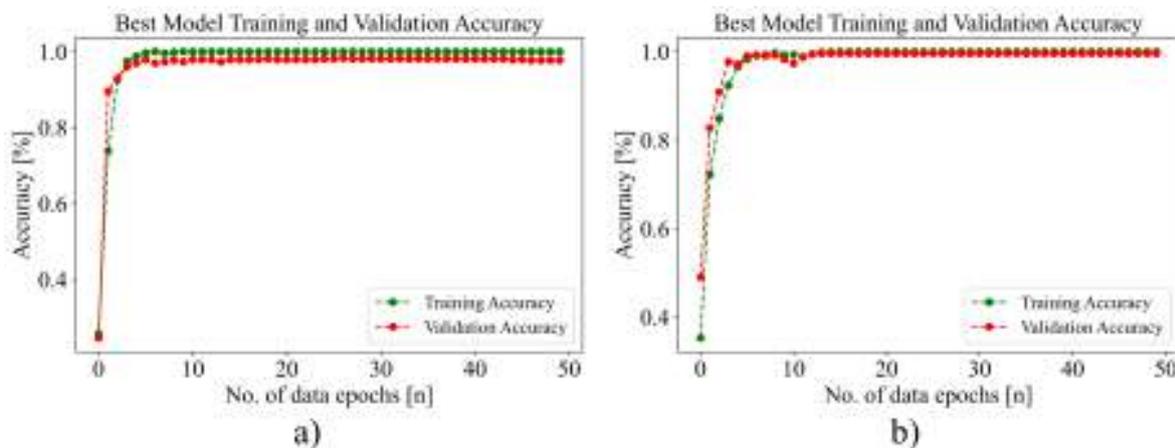


Figure 5.26: Best training accuracy and validation accuracy for: a) 1D multi-Kernel-DCNN-LSTM model, b) 2D multi-Kernel-DCNN-LSTM model.

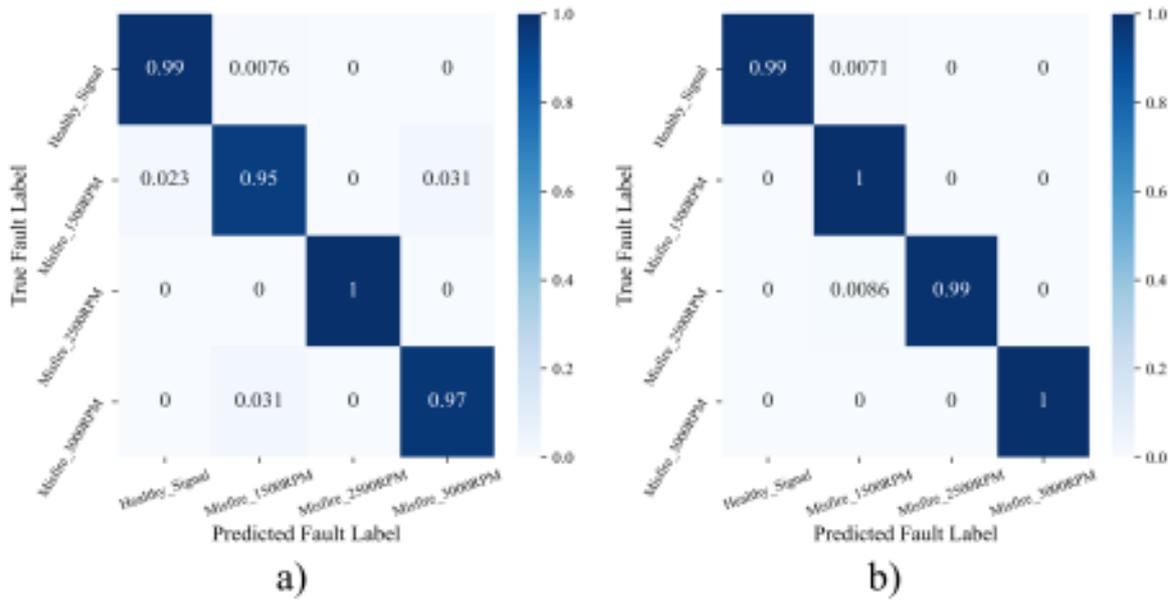


Figure 5.27: Confusion matrices representing true fault versus predicted fault for engine misfire at different RPMs: a) 1D multi-Kernel-DCNN-LSTM model, b) 2D multi-Kernel-DCNN-LSTM model.

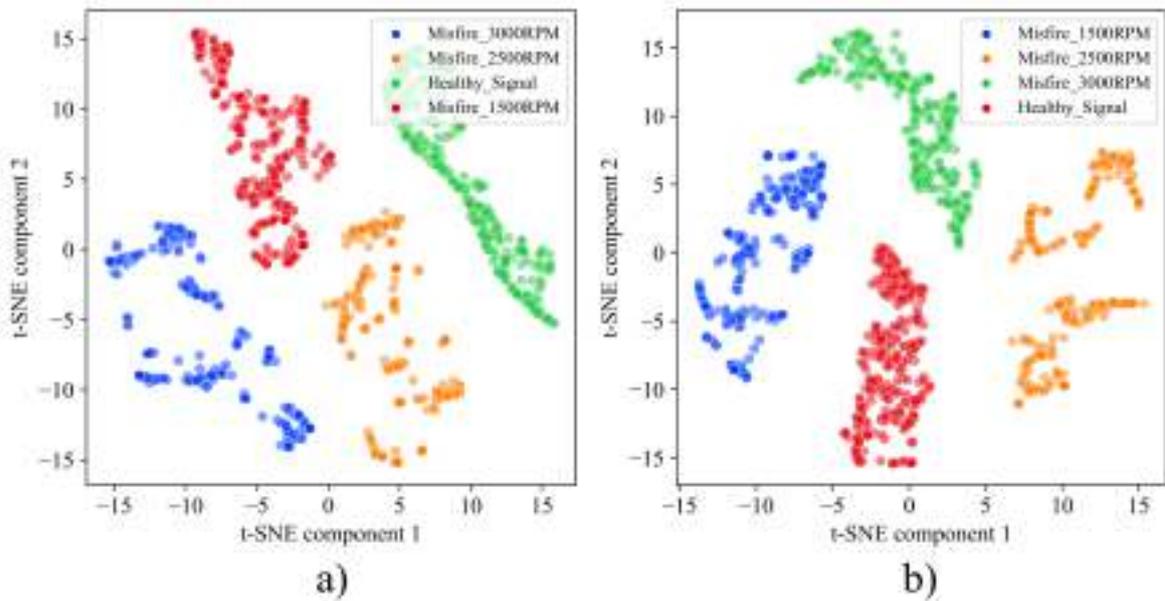


Figure 5.28: t-SNE visualization of vibration data after training: a) 1D multi-Kernel-DCNN-LSTM model, b) 2D multi-Kernel-DCNN-LSTM model.

Additionally, Figure 5.27 illustrates the confusion matrices for multi-Kernel-DCNN-LSTM models for both 1D and 2D datasets, providing a visual representation of the diagnostic accuracy for each health condition. According to the confusion matrices, 2D multi-Kernel-DCNN-LSTM possess the higher diagnostic accuracy for each health condition compared to the 1D multi-Kernel-DCNN-LSTM model. For 1D multi-Kernel-DCNN-LSTM, diagnostic accuracy is 99%, 95%, 100%, and 97%, whereas for 2D multi-

Kernel-DCNN-LSTM it is 99%, 100%, 98%, and 100% for healthy, misfire at 1500 RPM, misfire at 2500 RPM, and misfire at 3000 RPM, respectively.

Following the confusion matrices, Figure 5.28 provides t-SNE representation of the features embedding generated by the multi-Kernel-DCNN-LSTM models for both 1D and 2D datasets. These t-SNE visualization give insights into the distribution and

Table 5.9: Performance Results for 1D and 2D multi-Kernel-DCNN-LSTM Models.

Models	Health Condition	Accuracy	Sensitivity	Specificity	BA	GM
1D multi-Kernel DCNN-LSTM	Healthy Signal	99.23	99.24	99.22	99.23	99.23
	Misfire: 1500RPM	97.68	94.57	98.71	96.64	96.62
	Misfire: 2500RPM	100	100	100	100	100
	Misfire: 3000RPM	98.45	96.90	98.97	97.93	97.93
2D multi-Kernel DCNN-LSTM	Healthy Signal	99.82	99.29	100	99.65	99.64
	Misfire: 1500RPM	99.61	100	99.48	99.74	99.74
	Misfire: 2500RPM	99.78	99.14	100	99.57	99.57
	Misfire: 3000RPM	100	100	100	100	100

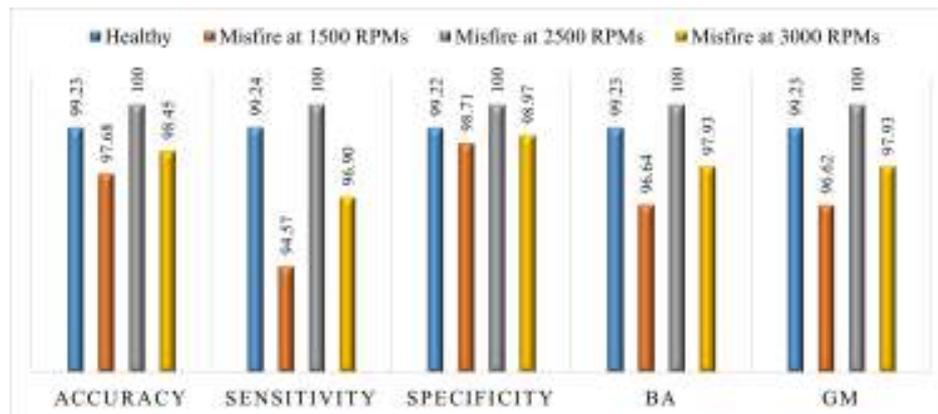


Figure 5.29: Performance results of 1D multi-Kernel-DCNN-LSTM model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.

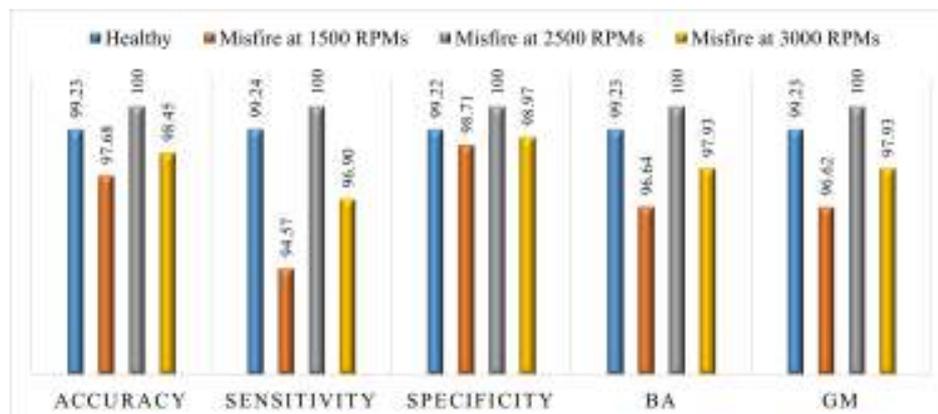


Figure 5.30: Performance results of 2D multi-Kernel-DCNN-LSTM model using vehicle misfire vibration dataset recorded using ADXL1002 accelerometer.

clustering of the extracted features.

Moreover, similarly to the previous section, Table 5.9 represents a detailed analysis of the conclusive performance results attained by our system for both 1D and 2D datasets for multi-Kernel-DCNN-LSTM models. To assess the system's efficacy comprehensively, multiple performance indicators similar to the previous section are discussed including accuracy, sensitivity, specificity, BA, and GM. These performance results are shown in the Figure 5.29 and Figure 5.30 for 1D and 2D datasets, respectively for visualization. The results illustrated in these figures prove the superiority of the 2D multi-Kernel-DCNN-LSTM model compared to the 1D multi-Kernel-DCNN-LSTM model.

5.7 Comparative Analysis of AI Methods

This study evaluated the performance of various AI models for diagnosing vehicle engine misfires using 1D time-domain data and 2D grayscale images. The models—1D DCNN, 1D DCNN-LSTM, 1D multi-Kernel-DCNN-LSTM, 2D DCNN, 2D DCNN-LSTM, and 2D multi-Kernel-DCNN-LSTM—were designed to analyze vibration signals recorded by the ADXL1002 accelerometer. By incorporating both 1D and 2D data representations, the analysis aimed to compare the diagnostic efficiency of these architectures in capturing patterns associated with engine conditions.

Figure 5.31 presents a comparative analysis of the maximum, minimum, and average accuracy achieved over 10 independent runs for different models using both 1D and 2D datasets. The results indicate that, in terms of the highest model accuracy, the 2D dataset outperforms the 1D dataset. Additionally, the multi-kernel DCNN-LSTM model achieves the highest accuracy among all evaluated models. The numerical values

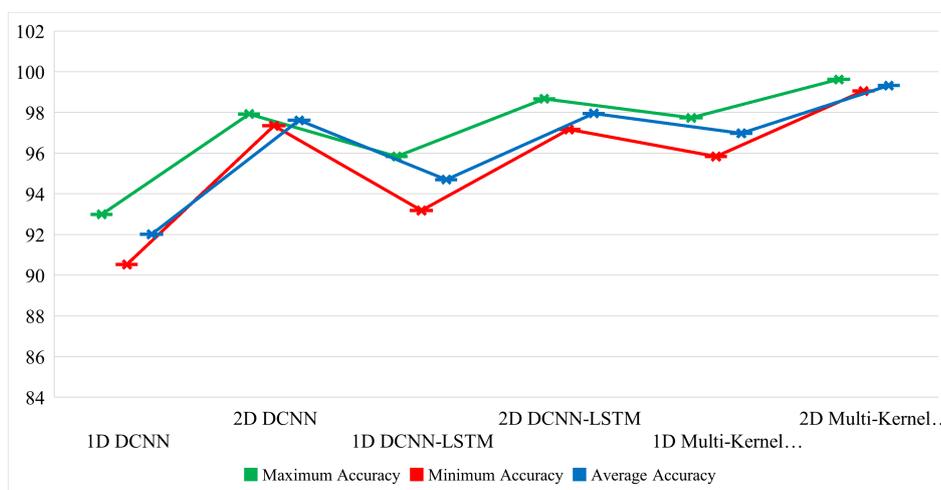


Figure 5.31: Comparison of maximum accuracy, minimum accuracy, and average accuracy for the 10 independent runs for different models with 1D and 2D datasets.

Table 5.10: Comparison of Maximum, Minimum, Average Accuracy, and Standard Deviation Across 10 Runs.

Model	Maximum Accuracy	Minimum Accuracy	Average Accuracy
1D DCNN	92.99	90.53	92.01
2D DCNN	97.92	97.35	97.61
1D DCNN-LSTM	95.83	93.18	94.70
2D DCNN-LSTM	98.67	97.16	97.95
1D Multi-Kernel DCNN-LSTM	97.73	95.83	96.97
2D Multi-Kernel DCNN-LSTM	99.62	99.05	99.32

corresponding to these accuracy metrics are provided in Table 5.10.

The 1D models focused on sequential data, while the 2D models utilized spatial representations, offering enhanced feature extraction and temporal analysis. This comprehensive approach facilitated an in-depth evaluation of each model’s diagnostic capabilities across varying engine states, including healthy and misfire conditions at different RPM levels (1500, 2500, and 3000).

5.7.1 Performance Analysis

Among the all tested models, multi-Kernel-DCNN-LSTM achieved the highest diagnostic accuracy over the hybrid DCNN-LSTM and simple DCNN models. Specifically 2D dataset helps to enhance the diagnostic accuracy over 1D dataset for all models. Hybrid DCNN-LSTM and multi-Kernel-DCNN-LSTM architectures effectively captured temporal dependencies and spatial features, emphasizing the importance of combining these attributes for fault detection. The 2D DCNN performed competitively but lacked the temporal modeling advantage, while the 1D DCNN exhibited relatively lower accuracy due to the absence of temporal processing.

Performance results for all models also show the superiority of the multi-Kernel-DCNN-LSTM models over the hybrid DCNN-LSMT and DCNN models. Confusion matrices further demonstrated the models’ proficiency in diagnosing misfire conditions across RPM levels. t-SNE plots provided additional insights into data separability, showcasing the models’ ability to distinguish healthy and faulty states. These findings validate the potential of 2D representations in enhancing the accuracy and robustness of misfire diagnosis systems.

5.7.2 Impact of 1D and 2D Data

One key finding is the significant improvement in diagnostic accuracy when using 2D representations over 1D inputs. The 2D data format, by preserving both temporal and

frequency information, enables the models to learn from spatially distributed features that are less apparent in 1D data. This advantage is particularly evident in the multi-Kernel-DCNN-LSTM model, where the combination of convolutional layers for spatial extraction and LSTM layers for temporal analysis leads to highly accurate classification across all misfire conditions.

The comparison underscores that for vibration-based fault diagnosis, 2D representations may be more suitable for capturing the intricate frequency components associated with misfire. The spatial features derived from the 2D grayscale images allow the models to detect subtle variations across different RPM levels, improving both the accuracy and robustness of fault diagnosis.

5.7.3 Challenges and Limitations

The results of this study suggest that employing a 2D multi-Kernel-DCNN-LSTM model with both spatial and temporal learning capabilities could be highly effective for real-world misfire detection in automotive settings. The model's ability to accurately diagnose misfire across multiple RPM levels demonstrates its robustness and potential for integration into on-board diagnostic systems. Furthermore, the use of low-cost sensors, such as the ADXL1002 accelerometer, combined with advanced AI models, makes this approach feasible and scalable for widespread adoption.

While the study demonstrates the potential of 2D representations and multi-Kernel-DCNN-LSTM models, there are limitations to consider. The transformation of vibration signals into 2D images adds a preprocessing step, which could increase computational requirements in real-time applications. Additionally, further research could explore the performance of these models across a broader range of fault types and engine conditions to enhance their generalizability.

5.8 Conclusion

The advancements in AI and their application in vehicle misfire fault diagnosis demonstrate a significant leap in addressing the limitations of traditional methods. This study meticulously explored the potential of deep learning architectures, including DCNN, hybrid DCNN-LSTM, and multi-Kernel-DCNN-LSTM models, in analyzing vibration data for fault detection. The findings emphasize the transformative role of AI in extracting and learning complex features from both 1D time-domain vibration data and 2D grayscale images, paving the way for more accurate and reliable diagnostic systems.

The comparative analysis of 1D and 2D data inputs revealed critical insights into their diagnostic efficacy. While 1D representations offered computational simplicity and

adequate temporal feature extraction, the richer feature space of 2D representations provided superior diagnostic accuracy, particularly in capturing frequency and spatial patterns indicative of misfire events. This distinction underscores the utility of 2D data for applications requiring detailed analysis of vibration signals, as seen in the improved performance of DCNN, hybrid DCNN-LSTM, and multi-Kernel-DCNN-LSTM models with 2D inputs.

Among the methodologies employed, the multi-Kernel-DCNN-LSTM model emerged as the most robust framework. By integrating the spatial feature extraction capabilities of CNN with the sequential learning strengths of LSTM, the model demonstrated exceptional accuracy across diverse operating conditions. The success of this architecture highlights the importance of leveraging both spatial and temporal features in diagnosing dynamic and intermittent faults like vehicle misfires. Notably, the proposed 2D multi-Kernel-DCNN-LSTM model achieved near-perfect classification accuracies, proving its efficacy for real-world applications.

The research further demonstrated the feasibility of using low-cost sensors such as the ADXL1002 accelerometer in conjunction with sophisticated AI models. This combination represents a promising approach for developing scalable and cost-effective diagnostic systems for the automotive industry. Moreover, the study's preprocessing pipeline, including techniques such as EMD, ensured the effective isolation of meaningful vibration features, enhancing model performance.

However, the study also acknowledged certain limitations. The transformation of raw vibration signals into 2D grayscale images introduces additional preprocessing steps that may pose challenges in real-time diagnostics. Moreover, while the proposed models showed robustness under varied conditions, their generalizability across other engine types and fault scenarios remains to be validated. Future research could focus on optimizing these architectures for real-time applications and exploring lightweight versions of the proposed models to ensure seamless integration into onboard diagnostic systems.

In conclusion, this chapter underscores the potential of AI-driven approaches in advancing the state of fault diagnosis in automotive engines. By successfully addressing the complexities of vibration signal analysis, this work lays a strong foundation for the development of intelligent, real-time diagnostic systems. The insights gained from this study not only highlight the strengths of AI in enhancing diagnostic accuracy but also open new avenues for innovation in vehicle health monitoring.

Chapter 6

Conclusions

The work presented in this dissertation provides a broad investigation into the successful integration, calibration, and practical deployment of the ADXL1002 capacitive microelectromechanical systems (MEMS) accelerometer with the Black BeagleBone controller. The study carefully details the hardware interfacing process, confirming accurate signal acquisition and optimal sensor performance for vibration-based condition monitoring. Following the successful calibration and validation of the sensor system, the research develops to its application in real-time vibration data acquisition for vehicle engine health diagnostics. Specifically, the study explores the development and implementation of advanced digital signal processing (DSP) techniques alongside artificial intelligence (AI)-based models to enable accurate fault diagnosis in vehicle engines and rolling element bearings.

A key aspect of this work is the evaluation of the success of low-cost MEMS-based ADXL1002 accelerometer, in acquiring and analyzing vibration signals for diagnostic purposes. The fundamental limitations of ADXL1002 accelerometers compared to the piezoelectric accelerometers, such as sensitivity and noise, are acknowledged in this research work. However, the developed signal processing and AI-driven methods for health diagnosis of vehicle engines and bearing faults validate that the ADXL1002 could be cost-effective and efficient health monitoring and fault diagnostic solutions in industrial and automotive applications.

The findings of this dissertation provide significant contributions to the field by raising understanding of the trade-offs between cost, precision, and reliability in vibration-based diagnostics. Furthermore, the research findings highlight both the benefits and limits of the proposed approaches, offering a comprehensive view of the practicality of using MEMS accelerometers in real-world fault identification conditions. Finally, this research provides the groundwork for future advances in the combination of DSP techniques with AI models to improve predictive maintenance and health surveillance of complex machines.

6.1 Interfacing and Calibration of the ADXL1002 Accelerometer

The ADXL1002 accelerometer was efficiently interfaced with the Black BeagleBone controller, allowing for the accurate capture of high-frequency vibration data. To assure the reliability and precision of the acquired vibration signals, a thorough calibration procedure was performed in a controlled laboratory setting. This approach required the use of a vibration shaker and a function generator to generate well-defined reference vibration signals. The vibration shaker was analytically activated at predefined frequencies, and the ADXL1002 sensor recorded the resulting vibration signals. The obtained sensor data was then rigorously compared to predetermined reference frequencies to determine the quality and consistency of the accelerometer's response. The findings of this comparative analysis show a strong correlation between the input reference signals and the ADXL1002's recorded readings, thereby proving the ADXL1002's accuracy and dependability for high-fidelity vibration detection. This calibration procedure was critical in ensuring that the experimental data acquired for defect diagnosis remained accurate and dependable. This approach increased confidence in the integrity of the obtained vibration signals, which is necessary for meaningful analysis in following stages of the study, by confirming the ADXL1002 accelerometer's functioning in a controlled context.

6.2 Vibration Data Acquisition and Preprocessing

Following calibration of ADXL1002 accelerometer, vibration data from a vehicle engine running under various regulated conditions was carefully recorded using the ADXL1002 accelerometer. In particular, three distinct engine speeds—1500 RPM, 2500 RPM, and 3000 RPM—were used to record vibration signals. These recordings included both regular (healthy) engine running as well as misfire conditions, allowing researchers to explore defect identification and diagnostics. In addition to engine vibration data, two datasets from rotating bearings subjected to varied early-stage fault circumstances were included; these bearing datasets are available in the literature, expanding the study's scope to include bearing problem detection.

To verify the reliability and integrity of the recorded vibration signals, empirical mode decomposition (EMD) was used as a preprocessing step. This technology was used to breakdown raw vibration signals into intrinsic mode functions (IMFs), successfully minimizing the effects of noise and external disturbances. This denoising procedure improved the quality of the datasets by removing the most pertinent signal components, making them more suitable for additional analysis.

After noise reduction, the refined vibration data were converted into two-dimensional

(2D) grayscale image representations. This conversion was critical in preparing data for AI-based diagnostic models since it allowed the use of deep learning techniques developed to process image-based features. The conversion of time-series signals to 2D picture representations accelerated the extraction of spatial and frequency-domain properties, which are required for accurate fault classification. Finally, this comprehensive preprocessing method was crucial in boosting the diagnostic performance of AI-driven models by assuring resilience and reliability in problem recognition across a variety of engine conditions.

6.3 Development of DSP Methods

The initial phase of the diagnostic framework focuses on the development and implementation of advanced DSP algorithms for extracting fault-related frequencies from vibration signals. This phase was crucial in developing a systematic method for assessing vibration data for defect diagnosis in bearings and vehicle engines. To accomplish this, several filtering algorithms were carefully created and tuned to improve the separation of distinctive fault frequencies in bearings, allowing them to be efficiently discriminated from other signal components. Metaheuristic techniques such as Harmony Search Algorithm (HAS) and Particle Swarm Optimization (PSO) were developed to optimize filter settings. HAS was combined with kurtosis, spectral kurtosis (SK), and short-time Fourier Transform (STFT)-based SK as fitness functions, while PSO was used with the Continuous Wavelet transform. Furthermore, an IMF-based FFT technique was developed for diagnosing vehicle engine misfires. The experimental results provided strong evidence that DSP-based approaches are extremely effective at diagnosing and recognizing fault states, particularly when the fault signatures have distinct spectral features and are well segregated from background noise. Despite their efficiency in controlled circumstances, the limitations of traditional DSP algorithms become apparent in more complex fault occurrences. When fault signs were weak, closely spaced, or obscured by noise and other frequency components, DSP approaches' capacity to deliver reliable fault diagnosis was limited. This demonstrated the importance of investigating more complex signal processing and machine learning technologies to increase diagnostic accuracy in difficult real-world scenarios.

6.4 Development of AI Methods

To address the key restrictions of classic DSP techniques in fault diagnosis, more advanced AI-based models were created and applied. Three deep learning architectures were investigated: a deep convolutional neural network (DCNN), a hybrid DCNN-long short-term memory (DCNN-LSTM) model, and an improved multi-Kernel DCNN-

LSTM model. These models were tested with two types of input representations: one-dimensional (1D) time-series data extracted directly from vibration signals and 2D grayscale images created from the same dataset. Comparative research revealed that the 2D image-based technique outperformed the 1D method in terms of diagnostic accuracy. This increase can be attributed to the capacity of 2D representations to preserve spatial patterns and feature correlations that are typically difficult to identify in time-domain inputs.

Furthermore, the hybrid DCNN-LSTM and multi-Kernel DCNN-LSTM models performed better than the standalone DCNN model. The use of LSTM layers in hybrid models allowed them to successfully capture temporal dependencies within sequential data, improving their ability to recognize fault-related patterns across time. Notably, the multi-Kernel DCNN-LSTM model scored the highest classification accuracy, thanks to its multi-Kernel framework, which allowed it to extract diverse and discriminative features from numerous receptive fields. This multi-scale feature extraction capability helped to improve its issue detection and diagnostic performance. Overall, our results highlight the effectiveness of AI-driven approaches in improving the accuracy and reliability of problem identification in vibration-based condition monitoring systems.

6.5 Comparative Analysis

The comparative analysis of diagnostic frameworks provides useful insights into the benefits and drawbacks of various health detection and diagnosis methods. Traditional DSP approaches were excellent in detecting basic faults, but they showed major limits when applied to more complicated and dynamic fault scenarios. In contrast, AI-based models, particularly the multi-Kernel-DCNN-LSTM architecture, have demonstrated significant accuracy and robustness in diagnosing problems under varied operating situations. This AI model's higher performance can be attributed to its capacity to extract and combine many elements from vibration data, resulting in a more thorough identification procedure. Notably, the multi-Kernel structure enables the model to capture both local and global signal features, making it particularly adept at recognizing intelligent fault signatures than conventional methods. These findings highlight the substantial power of AI-driven methods in vibration-based fault identification, especially when paired with advanced signal preprocessing techniques like EMD. The use of such techniques improves the reliability and precision of fault detection systems, paving the door for more complex and adaptable diagnostic frameworks in real-world applications.

6.6 Challenges and Advantages of the ADXL1002 Accelerometer

During this research, the ADXL1002 accelerometer was identified as a long-term and cost-effective solution for vibration measurement and problem diagnosis. Its low cost and compact form factor provide substantial advantages over traditional piezoelectric accelerometers, which, while known for their high precision and wide frequency response, are significantly more expensive and physically bulkier. These properties make the ADXL1002 particularly demanding for applications that require a balance of performance and cost-effectiveness. However, the deployment of this sensor presented various technological obstacles that needed to be addressed. One of the main issues was its susceptibility to noise and external disturbances, necessitating the use of thorough preprocessing techniques to ensure the integrity and trustworthiness of the recorded data. Furthermore, while the ADXL1002 was effective at recording vibration signals within a particular frequency range, its performance in high-frequency applications was lower to that of piezoelectric alternatives. Despite these constraints, the experimental results revealed that the ADXL1002 was capable of providing sufficiently accurate and dependable measurements for the applications explored in this study. As a result, this study emphasizes the value of MEMS-based accelerometers as a low-cost alternative for vibration monitoring.

6.7 Contributions and Implications of Research

This study makes several substantial advances in the field of vibration-based fault identification by focusing on critical issues and broadening the applicability of sophisticated diagnostic methods. First, it proves the potential of using low-cost capacitive MEMS accelerometers for high-frequency vibration monitoring, demonstrating that, despite their economical cost, these sensors can successfully capture critical vibration properties required for fault identification. This discovery broadens the range of possible applications for MEMS-based sensing technologies, particularly in industries where cost restrictions frequently limit the deployment of high-end instrumentation.

Second, this study presents a systematic and comprehensive framework for vibration signal preprocessing, which includes critical procedures including noise reduction, signal improvement, and feature extraction. The methodology assures that the retrieved characteristics are meaningful and resilient, increasing the dependability of future fault classification algorithms. This framework improves the accuracy of vibration-based diagnostics by correcting signal defects and enhancing data quality.

Third, the study contributes to the improvement of AI-driven diagnostic models by incorporating hybrid and multi-kernel designs, which provide better generalization and

adaptation to complicated fault patterns. The suggested AI models not only improve classification performance, but also present unique ways for dealing with variations in vibration signals, making them more useful in real-world applications.

Collectively, these advances have significant practical implications for industries that rely on condition monitoring and predictive maintenance, such as automotive, aerospace, and manufacturing. This research paves the door for increased operational efficiency, reduced downtime, and higher reliability in important mechanical systems by making fault diagnosis technologies more cost-effective, accurate, and scalable.

6.8 Limitations and Future Work

While the outcomes of this study provide substantial assurance in the field of vehicle health diagnosis, it is critical to note certain limits that necessitate additional research. One of the most significant issues is the performance of the ADXL1002 capacitive MEMS accelerometer, particularly in high-frequency and high-noise situations. Despite its noteworthy advantages, including as high sensitivity and wide bandwidth, its usefulness in such situations may be limited by signal attenuation, interference, and probable nonlinearity in response. To address these problems, additional investigation into complex sensor design, signal conditioning techniques, and filtering approaches is required to increase diagnostic accuracy and reliability.

Furthermore, while the AI models proposed in this study performed well within the context of the specified datasets, their capacity to generalize across varied operating situations, sensor configurations, and mechanical systems remains an open question. Extensive experimental testing with multiple datasets is required to evaluate the consistency of these models in various real-world circumstances, accounting for varying vehicle kinds, engine configurations, and external disturbances. Further study should focus on increasing dataset diversity and investigating transfer learning strategies to improve model adaptability and generalizability.

Another avenue for future work concerns the integration of additional sensor arrangements to complement the MEMS-based accelerometer used in this study. For instance, incorporating piezoelectric accelerometers, which typically offer higher signal fidelity in high-frequency domains, could provide a more comprehensive representation of vibration characteristics, thereby improving fault detection accuracy. Multi-sensor fusion methods could also be investigated to leverage complementary information from different sensing arrangements.

Furthermore, while the proposed AI-based diagnostic methods have shown promise in offline analysis, translating these methodologies into real-time diagnostic systems is crucial for practical implementation in industrial and automotive applications. Future research should prioritize the extension of computationally efficient models optimized

for embedded computing platforms. Real-time implementation would enable continuous condition monitoring, early fault detection, and predictive maintenance strategies, ultimately enhancing the reliability and operational efficiency of automotive systems.

Appendix A

ADXL1002: Performance, Bandwidth, and Features

A.1 Optimal Performance of the ADXL1002

For most applications, a $1\ \mu\text{F}$ capacitor is typically adequate to decouple the accelerometer from power supply noise, ensuring stable performance. Additionally, implementing a band-limiting filter at the output is essential to suppress out-of-band noise and unwanted signals. This helps in maintaining signal integrity and improving measurement accuracy. Additionally, an appropriate band-limiting filter at the output is crucial to suppress any out-of-band noise and signals, thereby enhancing the accuracy and reliability of the measurements [2]. A detailed description of each pin function is provided in Figure A.1.

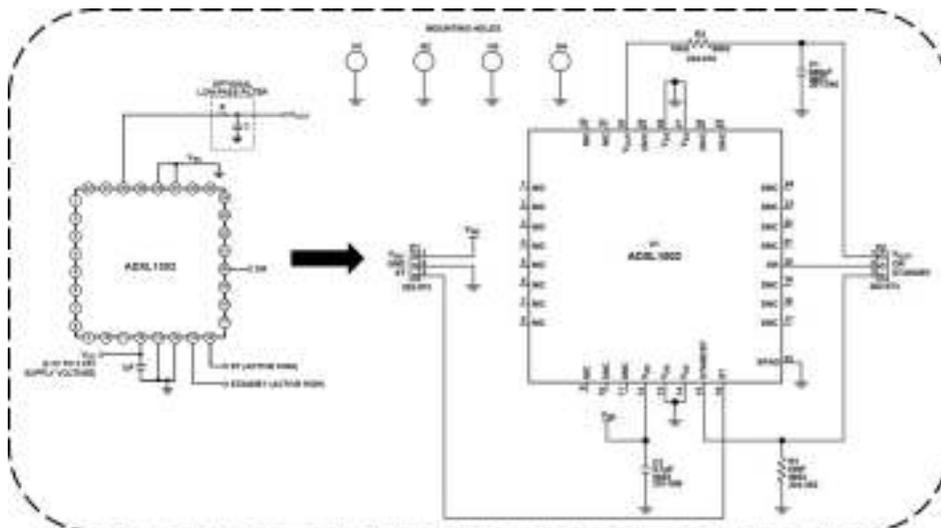


Figure A.1: Application circuit of ADXL1002 accelerometer.

The ADXL1002 accelerometer requires careful management of capacitive and resistive loads for optimal performance. It is recommended to use a capacitive load in the

range of 100 pF to 22 nF. The output amplifier of the accelerometer can drive resistive loads up to 2 mA of source current, which corresponds to resistances greater than 2.5 k Ω at 5 V operation. If the application demands driving a capacitive load equal to or greater than 100 pF, it is necessary to incorporate a series resistor of at least 8 k Ω to maintain amplifier stability. This ensures that the output signal remains stable and free from oscillations or distortions [2].

A.2 Operation of the ADXL1002

The ADXL1002 features several pins with specific functions that are crucial for its operation [2]:

- **ST (Self-Test) Pin:** This pin is used to activate the self-test feature, which is useful for verifying the functionality of the accelerometer. When the self-test function is not in use, this pin should be held low.
- **STANDBY Pin:** This pin allows the accelerometer to enter standby mode, significantly reducing power consumption. When the device is not actively measuring, the STANDBY pin should be held low to keep the accelerometer in its low-power state.
- **OR (Overrange Indicator) Pin:** This pin provides an indication when the measured acceleration exceeds the specified range of the accelerometer. Monitoring this output can help in assessing the system's status and ensuring that measurements remain within the operational limits.

A.3 Output Signal Bandwidth of the ADXL1002

The circuitry of the ADXL1002 supports an output signal bandwidth that extends beyond the resonant frequency of the sensor, enabling it to measure acceleration across a broad range. This broad range is comparable to the resonance of the ADXL1002 sensor. The output response is affected by both the sensor and the output amplifier, which makes it necessary to implement external band limiting or filtering. When operating beyond 10 kHz, several factors must be considered: the nonlinearity caused by the resonant frequency of the ADXL1002 sensor, increased noise from the wideband output of the amplifier, and discrete frequency spurious tones resulting from the internal 200 kHz clock coupling. Aliased interferers within the desired band cannot be eliminated, which can compromise observed performance. Therefore, high-speed sampling combined with appropriate band-limiting filtering is crucial for achieving optimal results [2, 27].

The ADXL1002's flat response bandwidth of 11 kHz, as depicted in Figure A.2, is ideally suited for frequency analysis up to 10 kHz. This characteristic ensures that

the accelerometer can accurately capture and analyze a wide range of vibration frequencies without significant attenuation or distortion. The sensor's resonant frequency, located at 21 kHz, indicates the point beyond which the sensor's response may become nonlinear and less reliable [2].

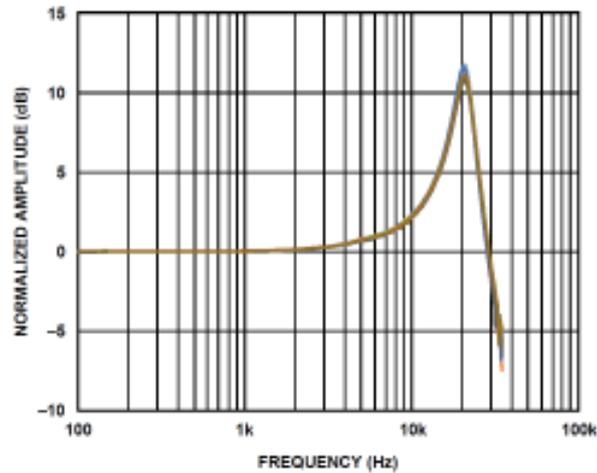


Figure A.2: Frequency response of ADXL1002 accelerometer [2].

The figure illustrates the frequency response curve of the ADXL1002, taken from the datasheet, highlighting its effectiveness in maintaining a consistent output within the specified bandwidth. The flat response up to 11 kHz means that the accelerometer can be used in applications requiring precise frequency analysis, such as machinery condition monitoring, where accurate detection of vibration frequencies is critical for diagnosing potential faults.

Beyond the flat response region, as the frequency approaches the sensor's resonant frequency of 21 kHz, the accelerometer's output may exhibit increased nonlinearity and higher noise levels. This necessitates careful consideration when analyzing vibrations in this higher frequency range. The data suggests that while the ADXL1002 can technically measure up to 21 kHz, optimal performance and accuracy are achieved within the 11 kHz bandwidth.

A.4 Features of the ADXL1002

The main features of the ADXL1002 accelerometer are as follows [2]:

- **Accelerometers:** analog
- **Axis:** X
- **Linear frequency response range:** from dc to 11 kHz (3 dB point)
- **Voltage supply:** 3.3 V to 5.25 V
- **Acceleration range:** ± 50 g
- **Noise density:** $25 \mu\text{g} / \sqrt{\text{Hz}}$

- **Sensitivity (mV/g):** 40
- **Bandwidth:** 21 kHz
- **Temperature operating range:** -40°C to +125°C

Appendix B

BeagleBone Black: Specifications and Component

B.1 I/Os of the BeagleBone Black

A key advantage of the BeagleBone Black is its extensive input/output (I/O) capabilities, which include digital I/O, analog inputs, PWM outputs, and multiple serial communication interfaces such as UART, I2C, and SPI. These features enable seamless integration with a variety of sensors, actuators, and other peripherals, making it possible to develop sophisticated embedded systems. Additionally, the board supports HDMI output, allowing for direct connection to displays, which is beneficial for graphical applications [89].

B.2 LEDs of the BeagleBone Black

The board includes four blue LEDs for user applications and a reset button to restart the processor. The microSD slot allows for the installation of a microSD card, and the USB client port, a mini-USB connection to a PC, can also supply power to the board. The BOOT switch is used to force the system to boot from the microSD card when the board is power-cycled. Storage needs are addressed by an onboard Micron eMMC (embedded MultiMediaCard) chip, offering storage capacities of up to 4GB. This integrated storage solution provides ample space for storing operating system files, applications, and user data, contributing to the board's standalone functionality.

B.3 Networking of the BeagleBone Black

Networking is another strength of the BeagleBone Black. The board includes a 10/100 Ethernet port and supports USB Wi-Fi adapters, facilitating the creation of In-

ternet of Things (IoT) applications. This enables the board to collect and process data from various sensors and communicate with cloud-based services, thereby enhancing its utility in modern connected applications. For network connectivity, the board employs the SMSC Ethernet PHY (Physical Layer Transceiver). This component provides the physical interface necessary for wired network connections, facilitating communication with other devices and network resources.

B.4 Power Management of the BeagleBone Black

Power management on the BeagleBone Black is handled by the TPS65217C PMIC (Power Management Integrated Circuit) [89]. This integrated circuit manages the distribution and regulation of power to various components on the board, ensuring stable and reliable operation even under varying power supply conditions. The power management design of the BeagleBone Black is both efficient and flexible. It can be powered via a USB connection or a 5V DC supply. Built-in mechanisms manage power fluctuations and ensure stable operation, and its low power consumption makes it ideal for portable and battery-powered projects. The primary DC power input is a 5V supply, and a power button signals the processor to initiate a power-down sequence.

B.5 Features of the BeagleBone Black

The main features of the BeagleBone Black are as follows [89]:

- **Memory:**
 - SDRAM: 512MB DDR3L 800MHZ
 - Onboard Flash: 4GB, 8bit Embedded MMC (eMMC)
- **SD/MMC Connector for microSD**
- **Power management:TPS65217C PMIC is used along with a separate LDO to provide power to the system**
- **Debug Support: Optional Onboard 20-pin CTI JTAG, Serial Header**
- **Power Source:**
 - miniUSB USB or DC Jack
 - 5VDC External Via Expansion Header
- **Connectivity:**
 - High speed USB 2.0 Client port: Access to USB0, Client mode via miniUSB
 - High Speed USB 2.0 Host port: Access to USB1, Type A Socket, 500mA LS/FS/HS
 - Serial Port: UART0 access via 6 pin 3.3V TTL Header. Header is populated
 - 10/100M Ethernet (RJ45)
- **User Input / Output:**

- o Reset Button
- o Boot Button
- o Power Button
- o LED power indicator
- o 4 user configurable LEDs
- **Video/Audio Interfaces:**
 - o HDMI D type interface
 - o LCD interface
 - o Stereo audio over HDMI interface
- **Expansion Interfaces via 2× 46 pin headers:**
 - o LCD, UART, eMMC
 - o ADC, I2C, SPI, PWM
- **Compact Size of a Mint Tin**
 - o The compact BeagleBone Black is small enough to fit into a mint tin

Appendix C

Bearing Fault Frequencies

The characteristic fault frequencies for rotating bearings are calculated using the following formulas:

C.1 Parameters

- N : Number of rolling elements.
- d : Diameter of rolling elements (ball diameter).
- D : Pitch diameter (distance from the center of the bearing to the center of the rolling elements).
- β : Contact angle (angle between the rolling elements and the raceways, in radians).
- f_r : Rotational frequency of the shaft (in Hz).

C.2 Fault Frequencies

1. **Ball Pass Frequency Outer Race (BPFO):**

$$f_{\text{BPFO}} = \frac{N}{2} f_r \left(1 - \frac{d}{D} \cos \beta \right)$$

2. **Ball Pass Frequency Inner Race (BPFIR):**

$$f_{\text{BPFIR}} = \frac{N}{2} f_r \left(1 + \frac{d}{D} \cos \beta \right)$$

3. **Ball Spin Frequency (BSF):**

$$f_{\text{BSF}} = \frac{D}{d} f_r \left[1 - \left(\frac{d}{D} \cos \beta \right)^2 \right]$$

4. **Fundamental Train Frequency (FTF):**

$$f_{\text{FTF}} = \frac{f_r}{2} \left(1 - \frac{d}{D} \cos \beta \right)$$

C.3 Steps to Compute

1. Identify bearing specifications: N , d , D , β , and f_r .
2. Substitute the values into the formulas to compute each fault frequency.
3. Analyze vibration spectra for peaks at these frequencies to identify specific faults.

Table C.1: Bearing Parameters.

Sr. no.	Parameters	IMF Dataset	MFPT Dataset
1	N	16	8
2	d	0.331	0.235
3	D	2.815	1.245
4	β	0.2648	0
5	f_r	2000/60	25

Appendix D

Converting 1D Data to 2D Grayscale Images

Below is the Python code divided into logical blocks, with explanations provided for each block.

D.1 Initialization

```
1 win_len = 4096 # 64x64 matrix size
2 stride = 2000 # Step size for overlapping segments
3
4 # Empty arrays to store data and labels
5 X = []
6 Y = []
```

Explanation: The window length (`win_len = 4096`) and stride (`stride = 2000`) were defined to control the segmentation of the time-domain data. An empty list `X` was initialized to store the 2D grayscale images, and `Y` was initialized to store the corresponding fault labels.

D.2 Iterating Through Fault Categories

```
1 for k in df['fault'].unique():
2     df_temp_2 = df[df['fault'] == k]
```

Explanation: The `for` loop iterates over each unique fault category in the dataset. For each fault type, a subset of the data (`df_temp_2`) was created by filtering rows that matched the current fault category. This ensures that images are generated separately for each fault type.

D.3 Segmenting and Reshaping

```
1 for i in np.arange(0, len(df_temp_2) - win_len, stride):
2     temp = df_temp_2.iloc[i:i + win_len, :-1].values
3     temp = temp.reshape((1, -1))
4     X.append(temp)
5     Y.append(df_temp_2.iloc[i + win_len, -1])
```

Explanation: For each fault type, the data was segmented into overlapping windows of length `win_len = 4096`, moving by `stride = 2000` samples. Each segment was then reshaped into a single row (1D array) using `reshape((1, -1))` and appended to `X`. The corresponding fault label was extracted and appended to `Y`. This ensured that each windowed segment of data was linked to its respective fault type.

D.4 Reshaping 1D to 2D

```
1 X = np.array(X)
2 X = X.reshape((X.shape[0], 64, 64))
```

Explanation: The collected 1D segments in `X` were reshaped into 64×64 2D matrices. Each segment of 4096 samples was split into 64 rows of 64 samples, with the first 64 samples forming the first row, the next 64 samples forming the second row, and so on, until all 4096 samples were used.

The resulting 64×64 matrices were treated as grayscale images, with each matrix entry corresponding to the pixel intensity. The amplitude values of the vibration signal directly determined the grayscale intensity, enabling a visual representation of the signal's variation.

D.5 Label Encoding and One-Hot Encoding

```
1 Y = np.array(Y)
2 encoder = LabelEncoder()
3 encoder.fit(Y)
4 encoded_Y = encoder.transform(Y)
5 OHE_Y = to_categorical(encoded_Y)
```

Explanation: The fault labels stored in `Y` were converted into numerical format using the `LabelEncoder`. This ensured that categorical labels (e.g., fault types) were mapped to integer values. These integers were then converted into one-hot encoded vectors using `to_categorical`, a format suitable for classification tasks in neural networks.

D.6 Summary of Workflow

- **Block 1:** Defined parameters and initialized storage for data and labels.
- **Block 2:** Filtered the dataset for each fault category.
- **Block 3:** Segmented 1D data into overlapping windows and stored them alongside their labels.
- **Block 4:** Converted the 1D windows into 64×64 grayscale images by reshaping.
- **Block 5:** Encoded the fault labels into one-hot vectors for use in machine learning models.

This modular approach clearly separates the steps of the 1D-to-2D transformation and prepares the data for deep learning applications.

Appendix E

t-SNE Implementation

The following Python code applies t-distributed Stochastic Neighbor Embedding (t-SNE) for dimensionality reduction on a vibration dataset with 1760 signals, each having 4096 data points. The resulting 2D embeddings are visualized as a scatter plot, where clusters represent different fault types. Each block of code is explained in detail below.

E.1 Load dataset and ensure correct data type

```
1 X_pre_aen = np.asarray(X, dtype='float64')
```

Explanation: X represents the original vibration dataset with shape (1760, 4096) and `np.asarray(X, dtype='float64')` converts it into a NumPy array with 64-bit floating-point precision to ensure compatibility with t-SNE.

E.2 Apply t-SNE for dimensionality reduction

```
1 m = TSNE(n_components, perplexity, learning_rate, verbose)
2 X_t_sne = m.fit_transform(X_pre_aen) # Reduce from 4096 to 2
```

Explanation: TSNE function from `sklearn.manifold` is used to reduce the dataset from 4096 dimensions to 2 using the following parameters and `fit_transform` fits the t-SNE model to the data and generates a new dataset `X_t_sne` with shape (1760, 2).

- `n_components=2`: Reduces the dataset to two dimensions.
- `perplexity=40`: Balances local and global data structure by considering approximately 40 nearest neighbors for each point.
- `learning_rate=100`: Controls the optimization step size for embedding updates.
- `verbose=1`: Displays progress updates (e.g., iterations and KL divergence).

E.3 Create a DataFrame for visualization

```
1 tSNEdf = pd.DataFrame(data=X_t_sne, columns=['t-SNE component 1', 't-SNE component 2'])
2 tSNEdf['Fault'] = Y # Add fault labels to the DataFrame
```

Explanation: `pd.DataFrame` creates a DataFrame with two columns for the t-SNE components whereas `tSNEdf['Fault'] = Y` adds a column containing fault labels (Y) to the DataFrame for color-coding the points in the plot.

E.4 Plot the 2D t-SNE components

```
1 # Step 4: Plot the 2D t-SNE components
2 fig_tSNE_1D_LSTM, ax = plt.subplots(figsize=(10, 10))
3 sns.scatterplot(x=tSNEdf['t-SNE component 1'],
4                 y=tSNEdf['t-SNE component 2'],
5                 hue='Fault',
6                 data=tSNEdf,
7                 palette="bright",
8                 alpha=0.5)
```

Explanation: `sns.scatterplot` generates a scatter plot with x-axis and y-axis where x is first t-SNE component (t-SNE component 1) and y is second t-SNE component (t-SNE component 2). Furthermore, each point is colored based on their fault labels with `hue='Fault'` and `palette="bright"`. `alpha=0.5` sets the transparency of the points.

Appendix F

Accuracy and Loss of 10 Runs

The proposed AI models including DCNN, hybrid DCNN-LSTM, and multi-Kernel-DCNN-LSTM models were proposed for both 1D and 2D datasets to diagnose the health conditions of vehicle engine using the low-cost MEMS based ADXL1002 accelerometer. The models were verified using the 10 independent runs for each dataset. Following figures show the output results for the training accuracy, training loss, validation accuracy, and validation loss for all 10 runs.

F.1 DCNN model for 1D and 2D dataset

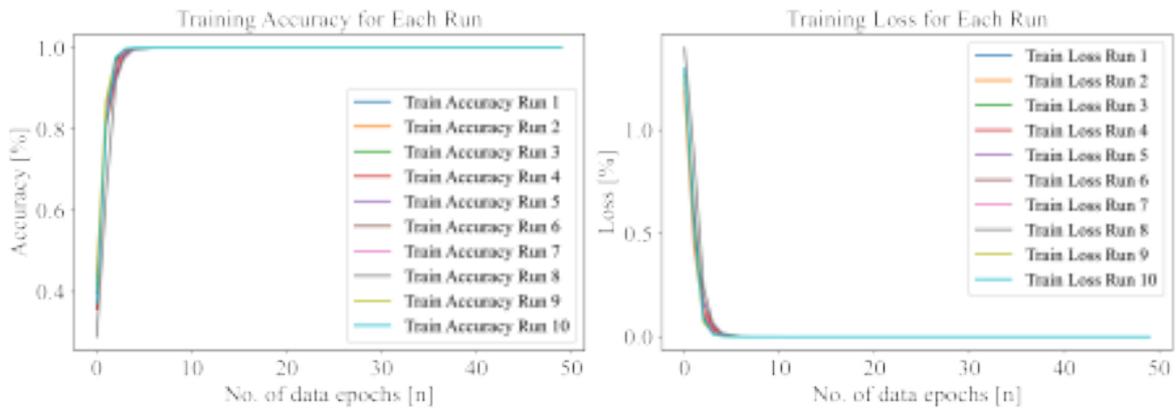


Figure F.1: Training accuracy and training loss of 10 runs using 1D DCNN model.

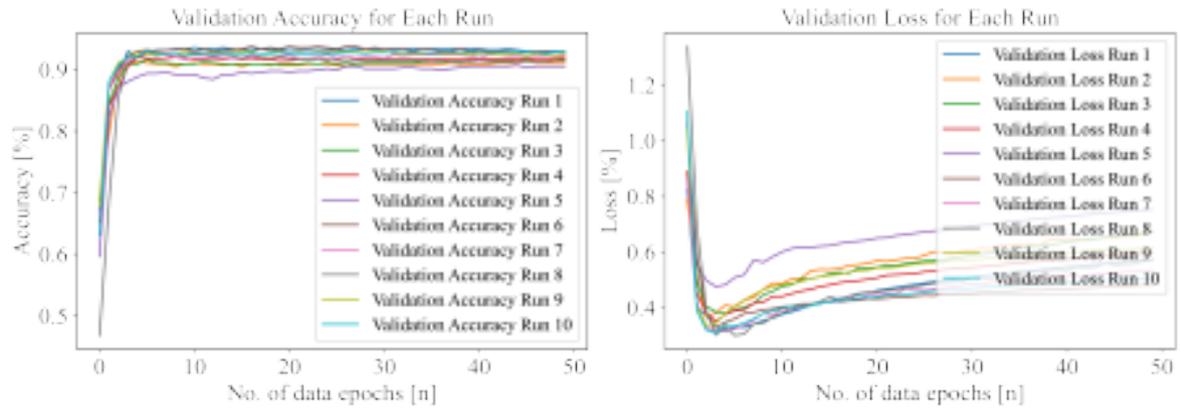


Figure F.2: Validation accuracy and validation loss of 10 runs using 1D DCNN model.

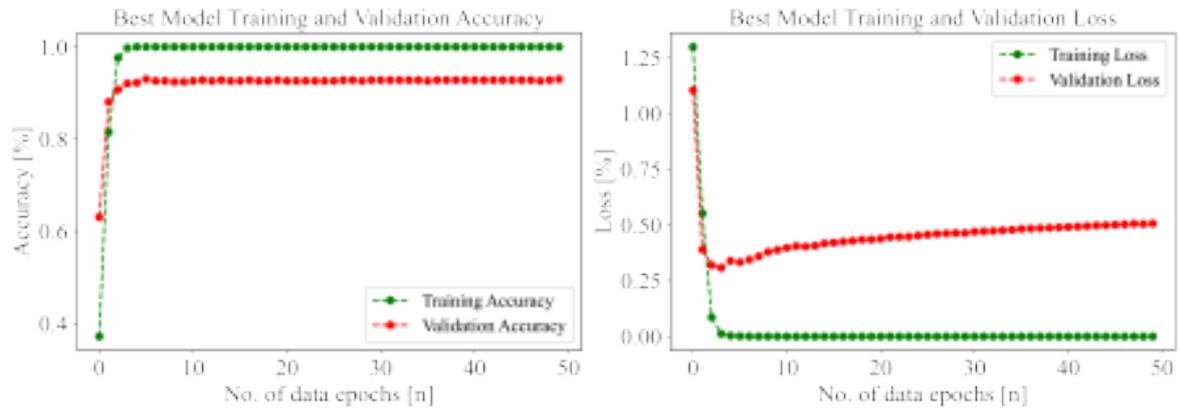


Figure F.3: Best model training and validation accuracy and loss for 1D DCNN model.

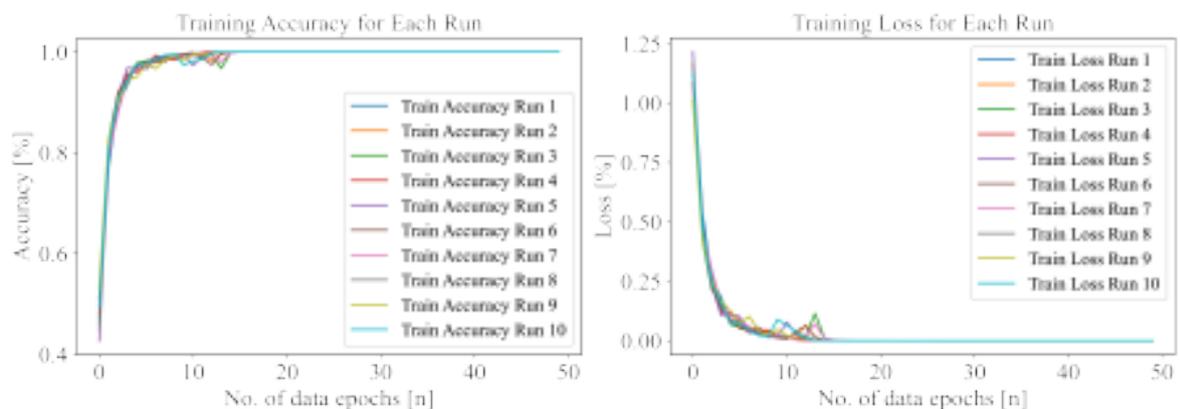


Figure F.4: Training accuracy and training loss of 10 runs using 2D DCNN model.

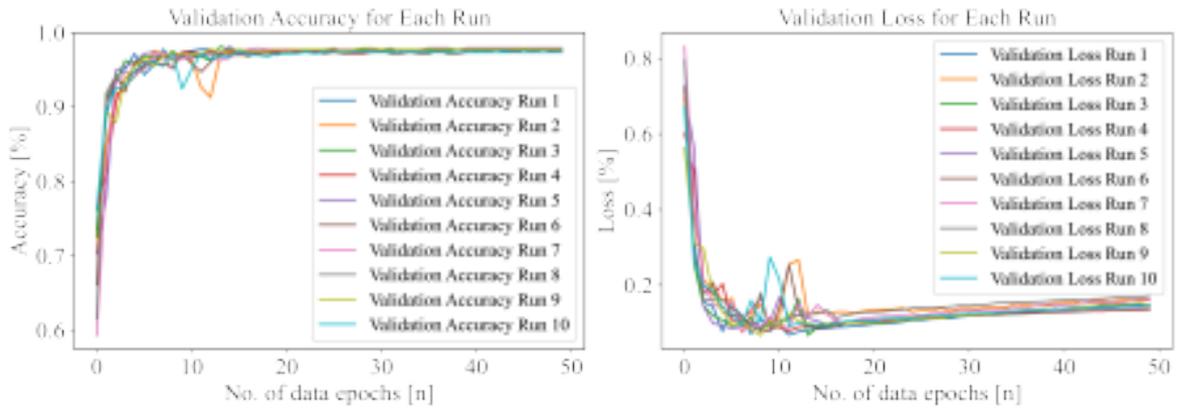


Figure F.5: Validation accuracy and validation loss of 10 runs using 2D DCNN model.

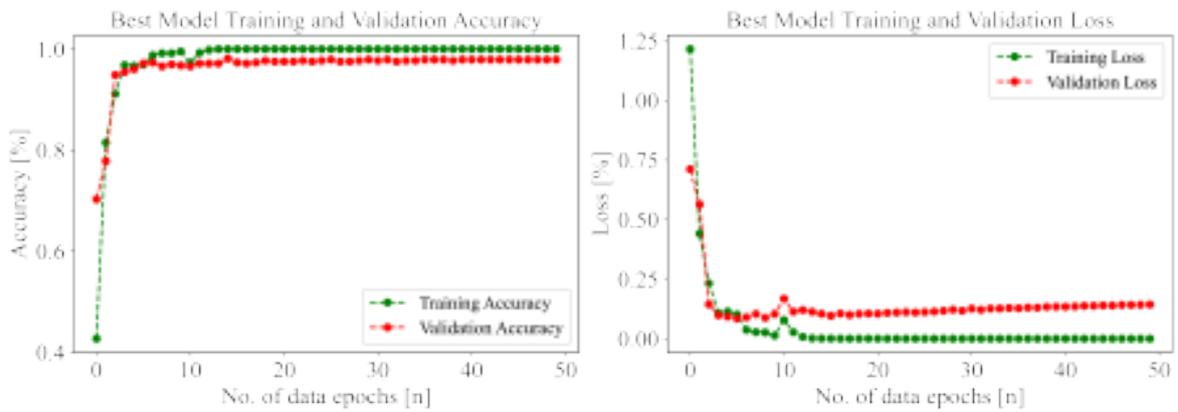


Figure F.6: Best model training and validation accuracy and loss for 2D DCNN model.

F.2 DCNN-LSTM model for 1D and 2D dataset

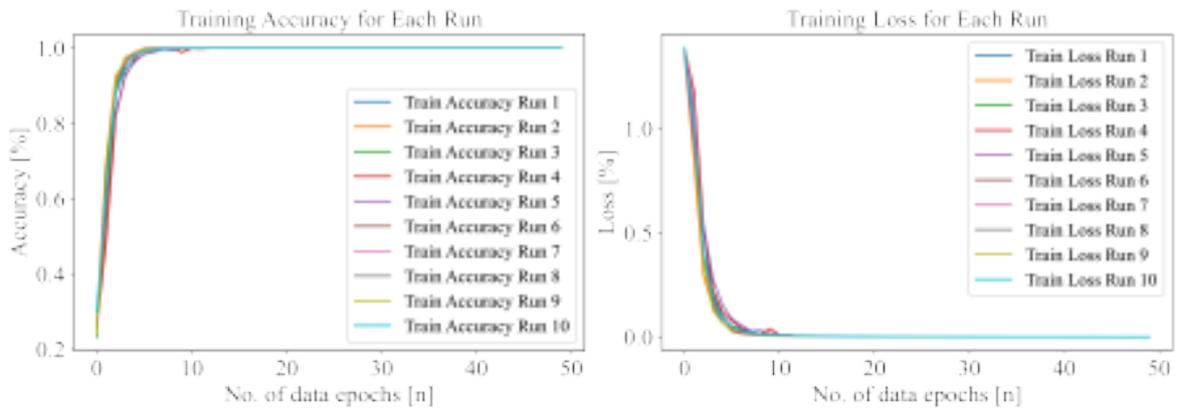


Figure F.7: Training accuracy and training loss of 10 runs using 1D DCNN-LSTM model.

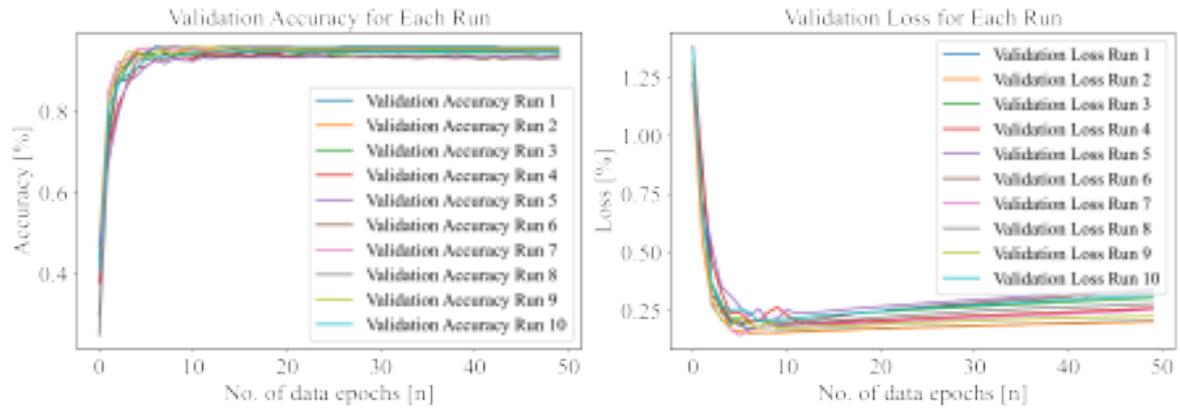


Figure F.8: Validation accuracy and validation loss of 10 runs using 1D DCNN-LSTM model.

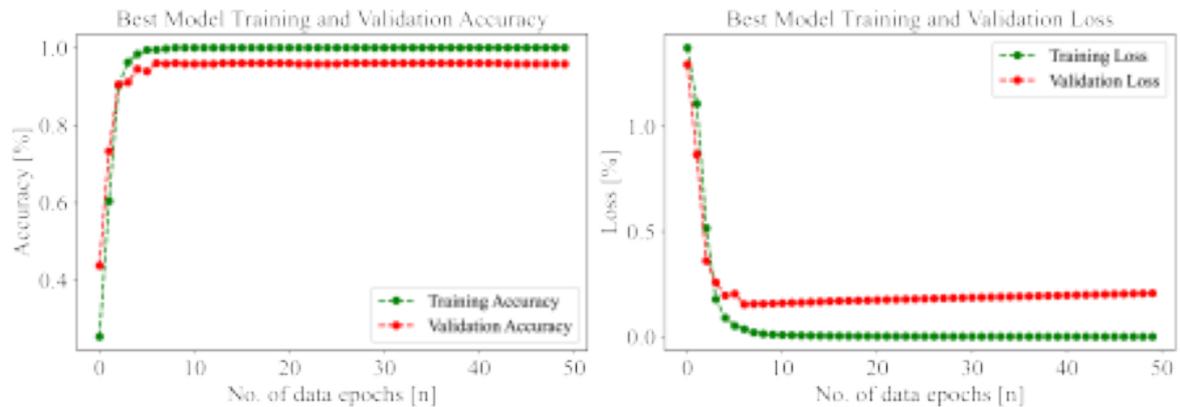


Figure F.9: Best model training and validation accuracy and loss for 1D DCNN-LSTM model.

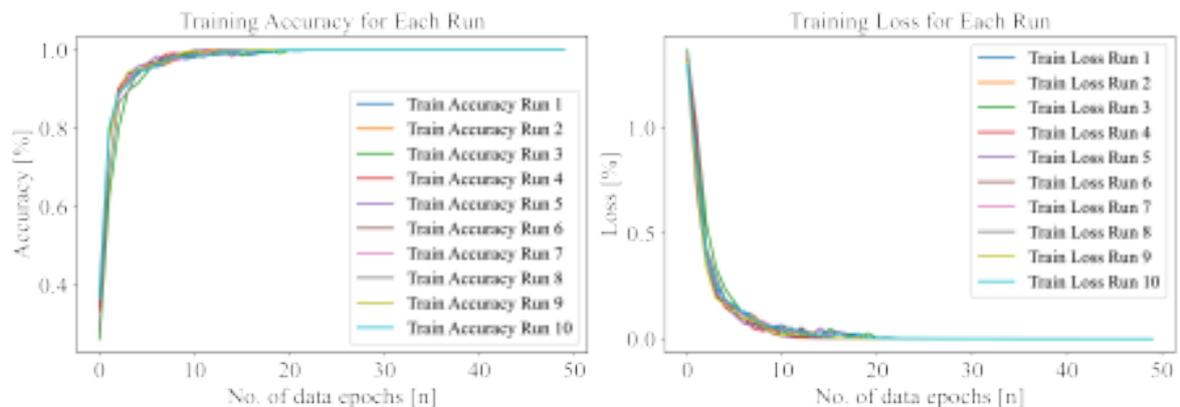


Figure F.10: Training accuracy and training loss of 10 runs using 2D DCNN-LSTM model.

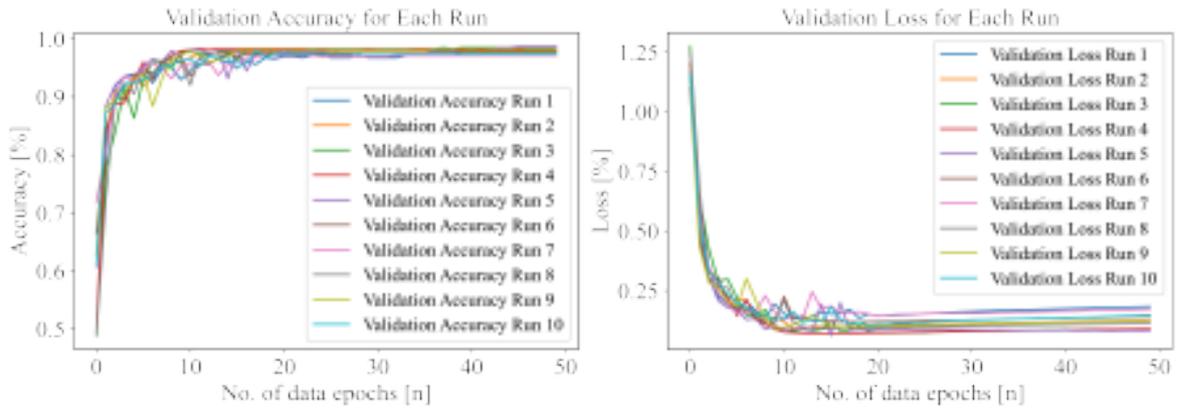


Figure F.11: Validation accuracy and validation loss of 10 runs using 2D DCNN-LSTM model.

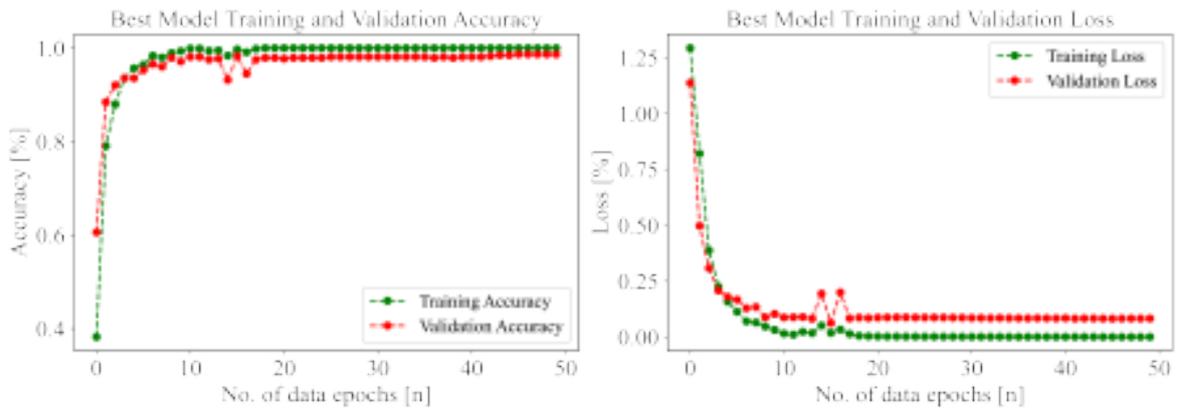


Figure F.12: Best model training and validation accuracy and loss for 2D DCNN-LSTM model.

F.3 Multi-Kernel DCNN-LSTM model for 1D and 2D dataset

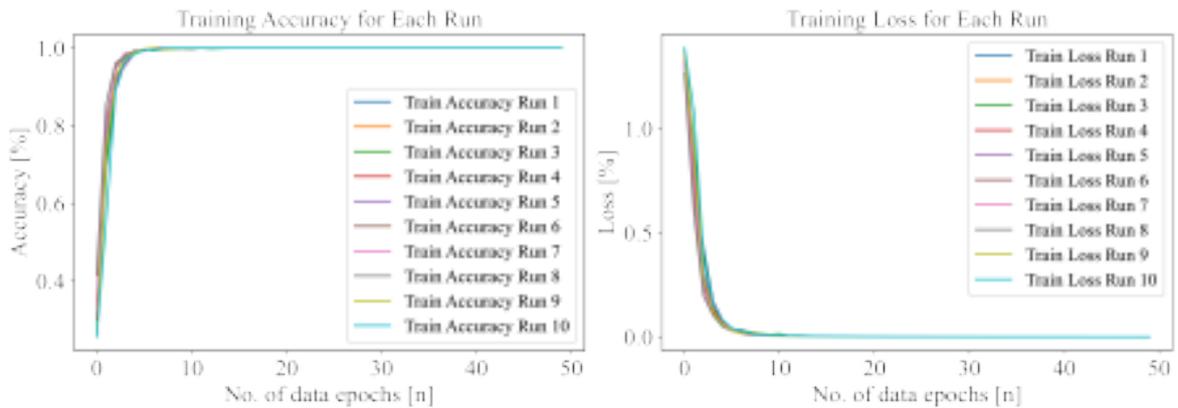


Figure F.13: Training accuracy and training loss of 10 runs using 1D multi-Kernel DCNN-LSTM model.

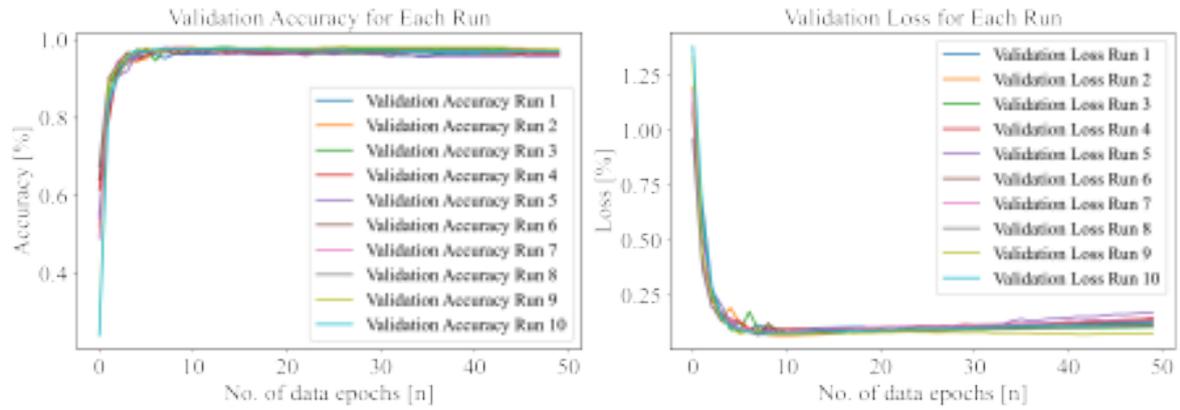


Figure F.14: Validation accuracy and validation loss of 10 runs using 1D multi-Kernel DCNN-LSTM model.

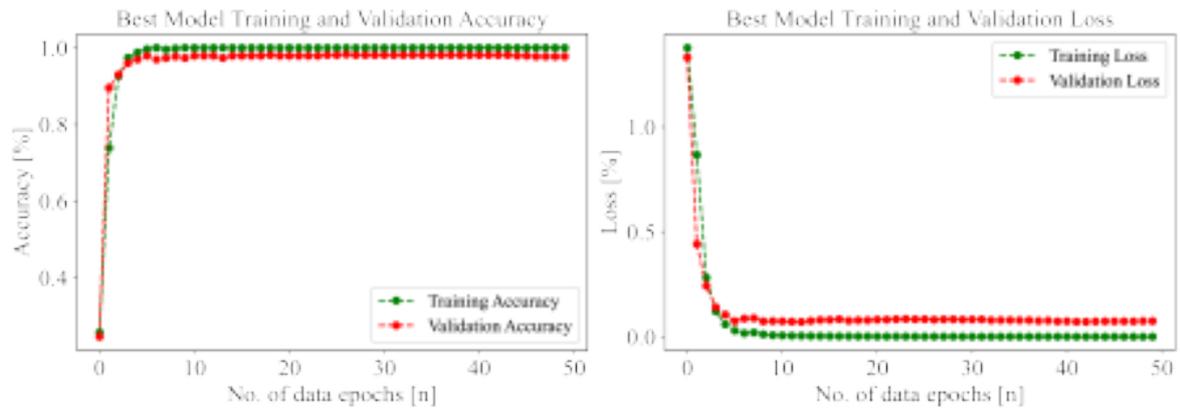


Figure F.15: Best model training and validation accuracy and loss for 1D multi-Kernel DCNN-LSTM model.

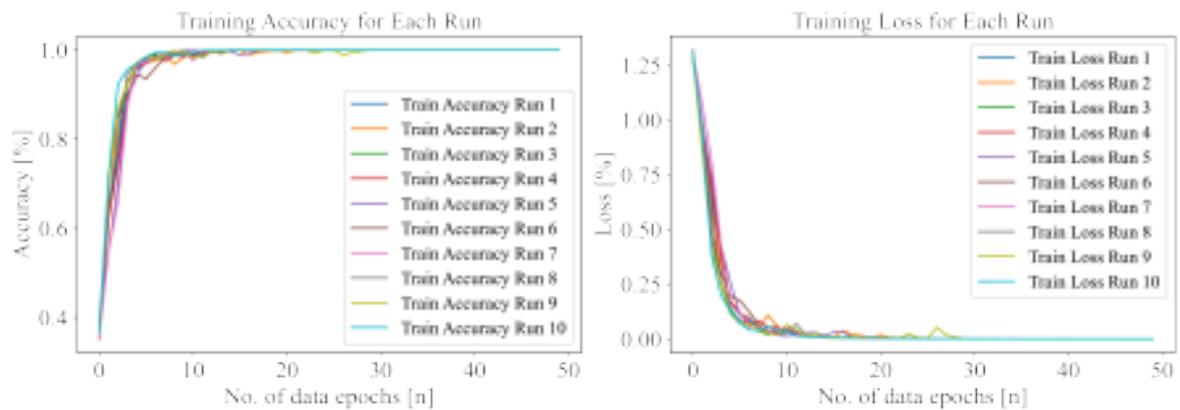


Figure F.16: Training accuracy and training loss of 10 runs using 2D multi-Kernel DCNN-LSTM model.

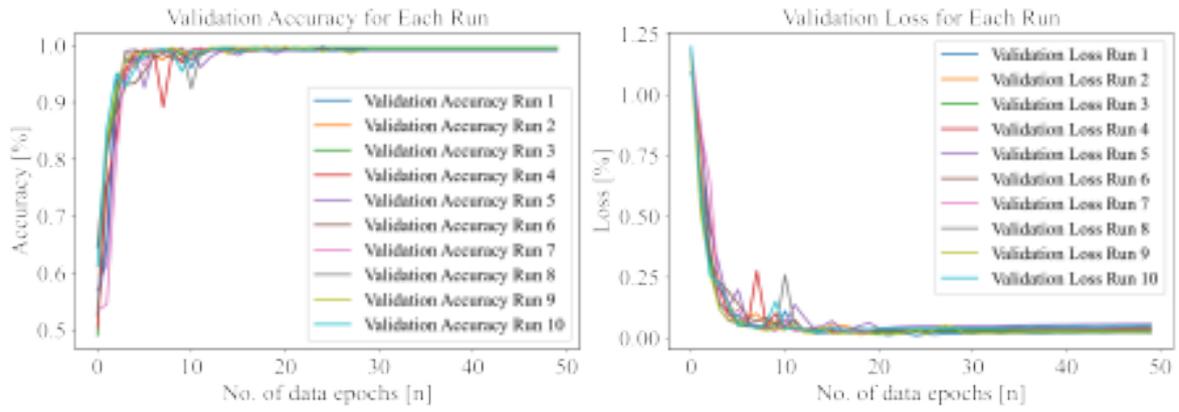


Figure F.17: Validation accuracy and validation loss of 10 runs using 2D multi-Kernel DCNN-LSTM model.

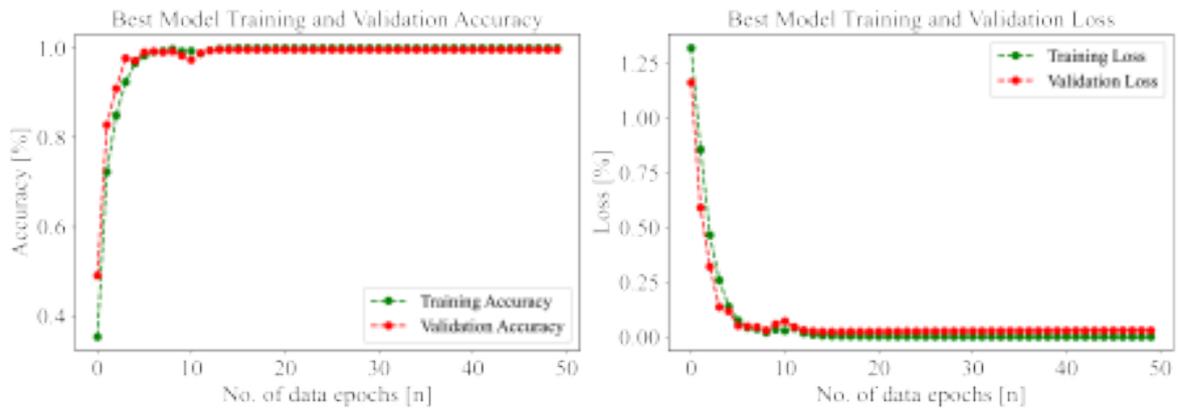


Figure F.18: Best model training and validation accuracy and loss for 2D multi-Kernel DCNN-LSTM model.

Bibliography

- [1] Hai Qiu, Jay Lee, Jing Lin, and Gang Yu. Wavelet filter-based weak signature detection method and its application on rolling element bearing prognostics. *Journal of Sound and Vibration*, 289(4):1066–1090, 2006.
- [2] Analog Devices. Adxl1002 datasheet. <https://www.analog.com/en/products/adxl1002.html#part-details>. Accessed: 2024-11-16.
- [3] OICA. International organization of motor vehicle manufacturers, 1919. Accessed: 2024-05-20.
- [4] Khodadadi Ali, Ghandiparsi Soroush, and Chuah Chen-Nee. A natural language processing and deep learning based model for automated vehicle diagnostics using free-text customer service reports. *Machine Learning with Applications*, 10:100424, 2022.
- [5] Naeem Hossain, Mustafizur Rahman, and Devarajan Ramasamy. Artificial intelligence-driven vehicle fault diagnosis to revolutionize automotive maintenance: A review. *CMES - Computer Modeling in Engineering and Sciences*, 141(2):951–996, 2024.
- [6] Md. Abdur Rahim, Md. Arafatur Rahman, Md. Mustafizur Rahman, Nafees Zaman, Nour Moustafa, and Imran Razzak. An intelligent risk management framework for monitoring vehicular engine health. *IEEE Transactions on Green Communications and Networking*, 6(3):1298–1306, 2022.
- [7] Meghdad Khazaei, Ahmad Banakar, Barat Ghobadian, Mostafa Agha Mirsalim, and Saeid Minaei. Remaining useful life (rul) prediction of internal combustion engine timing belt based on vibration signals and artificial neural network. *Neural Computing and Applications*, 33(13):7785–7801, 2021.
- [8] Syed Maaz Shahid, Sunghoon Ko, and Sungoh Kwon. Real-time abnormality detection and classification in diesel engine operations with convolutional neural network. *Expert Systems with Applications*, 192:116233, 2022.

- [9] Anil Kumar, C.P. Gandhi, Yuqing Zhou, Govind Vashishtha, Rajesh Kumar, and Jiawei Xiang. Improved cnn for the diagnosis of engine defects of 2-wheeler vehicle using wavelet synchro-squeezed transform (wsst). *Knowledge-Based Systems*, 208:106453, 2020.
- [10] Yixin Huangfu, Essam Seddik, Saeid Habibi, Alan Wassying, and Jimi Tjong. Fault detection and diagnosis of engine spark plugs using deep learning techniques. *SAE International Journal of Engines*, 15(4):515–525, nov 2021.
- [11] Xiufeng Li, Ning Wang, Yelin Lyu, Yan Duan, and Jiaqi Zhao. Data-driven fault early warning model of automobile engines based on soft classification. *Electronics*, 12(3), 2023.
- [12] Chengjin Qin, Yanrui Jin, Zhinan Zhang, Honggan Yu, Jianfeng Tao, Hao Sun, and Chengliang Liu. Anti-noise diesel engine misfire diagnosis using a multi-scale cnn-lstm neural network with denoising module. *CAAI Transactions on Intelligence Technology*, 8(3):963–986, 2023.
- [13] Leonardo Tomasi Francis, Victor Elízio Pierozan, Giovani Gracioli, and Gustavo Medeiros de Araujo. Data-driven anomaly detection of engine knock based on automotive ecu. In *2022 XII Brazilian Symposium on Computing Systems Engineering (SBESC)*, pages 1–8, 2022.
- [14] Mengqi Yang, Hui Chen, and Cong Guan. Research on diesel engine fault diagnosis method based on machine learning. In *2022 4th International Conference on Frontiers Technology of Information and Computer (ICFTIC)*, pages 1078–1082, 2022.
- [15] Dimitrios Rimpas, Andreas Papadakis, and Maria Samarakou. Obd-ii sensor diagnostics for monitoring vehicle operation and consumption. *Energy Reports*, 6:55–63, 2020. *Technologies and Materials for Renewable Energy, Environment and Sustainability*.
- [16] Hwa-seon Kim, Seong-jin Jang, and Jong-wook Jang. A study on development of engine fault diagnostic system. *Mathematical Problems in Engineering*, 2015(1):271374, 2015.
- [17] Pooja Rajendra Sawant and Yashwant B Mane. Design and development of on-board diagnostic (obd) device for cars. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–4, 2018.

- [18] Atef Hmida, Ahmed Hammami, Fakher Chaari, Mounir Ben Amar, and Mohamed Haddar. Effects of misfire on the dynamic behavior of gasoline engine crankshafts. *Engineering Failure Analysis*, 121:105149, 2021.
- [19] Abhishek Sharma, V. Sugumaran, and S. Babu Devasenapati. Misfire detection in an ic engine using vibration signal and decision tree algorithms. *Measurement*, 50:370–380, 2014.
- [20] S. Naveen Venkatesh, G. Chakrapani, S. Babudeva Senapti, K. Annamalai, M. Elangovan, V. Indira, V. Sugumaran, and Vetri Selvi Mahamuni. Misfire detection in spark ignition engine using transfer learning. *Computational Intelligence and Neuroscience*, 2022:1 – 13, 2022.
- [21] Canyi Du, Fei Jiang, Kang Ding, Feng Li, and Feifei Yu. Research on feature extraction method of engine misfire fault based on signal sparse decomposition. *Shock and Vibration*, 2021:1 – 12, 2021.
- [22] Jianfeng Tao, Chengjin Qin, Weixing Li, and Chengliang Liu. Intelligent fault diagnosis of diesel engines via extreme gradient boosting and high-accuracy time–frequency information of vibration signals. *Sensors*, 19(15), 2019.
- [23] João L. Firmino, João M. Neto, Andersson G. Oliveira, José C. Silva, Koje V. Mishina, and Marcelo C. Rodrigues. Misfire detection of an internal combustion engine based on vibration and acoustic analysis. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 47:1 – 12, 2021.
- [24] Mohamad Hazwan Mohd Ghazali and Wan Rahiman. Vibration analysis for machine monitoring and diagnosis: A systematic review. *Shock and Vibration*, 2021(1):9469318, 2021.
- [25] Morten Opprud Jakobsen, Eskild Sune Herskind, Kim Bjerger, Peter Ahrendt, Christian Fischer Pedersen, and Mikael Bergholz Knudsen. Vibration signatures in ball bearings as a function of lubricant viscosity ratio k , under alternating lubrication conditions. *Tribology International*, 156:106840, 2021.
- [26] Morten Opprud Jakobsen, Eskild Sune Herskind, Christian Fischer Pedersen, and Mikael Bergholz Knudsen. Detecting insufficient lubrication in rolling bearings, using a low cost mems microphone to measure vibrations. *Mechanical Systems and Signal Processing*, 200:110553, 2023.
- [27] Morten Opprud Jakobsen. Low cost mems accelerometer and microphone based condition monitoring sensor, with lora and bluetooth low energy radio. *HardwareX*, 18:e00525, 2024.

- [28] Hossein R. Najafabadi, Tiago G. Goto, Thiago C. Martins, Marcos S.G. Tsuzuki, and Ahmad Barari. Designing mems accelerometer for enhanced sensitivity and reduced cross-sensitivity in landslide monitoring. *Measurement*, 226:114092, 2024.
- [29] Huifang Xiao, Zedong Li, Xin Zhang, Liming Wang, Hamideh Khanbareh, and Chris Bowen. Mathematic modeling of vibration transmission path with discontinuous contact interfaces for gear dynamics in the gear-shaft-bearing-housing system. *Measurement*, 226:114054, 2024.
- [30] Abdenour Soualhi, Kamal Medjaher, and Nouredine Zerhouni. Bearing health monitoring based on hilbert–huang transform, support vector machine, and regression. *IEEE Transactions on Instrumentation and Measurement*, 64(1):52–62, 2015.
- [31] Samir Khan and Takehisa Yairi. A review on the application of deep learning in system health management. *Mechanical Systems and Signal Processing*, 107:241–265, 2018.
- [32] Xiang Li, Wei Zhang, Nan-Xi Xu, and Qian Ding. Deep learning-based machinery fault diagnostics with domain adaptation across sensors at different places. *IEEE Transactions on Industrial Electronics*, 67(8):6785–6794, 2020.
- [33] Junbin Chen, Ruyi Huang, Kun Zhao, Wei Wang, Longcan Liu, and Weihua Li. Multiscale convolutional neural network with feature alignment for bearing fault diagnosis. *IEEE Transactions on Instrumentation and Measurement*, 70:1–10, 2021.
- [34] Yuqing Zhou, Gaofeng Zhi, Wei Chen, Qijia Qian, Dedao He, Bintao Sun, and Weifang Sun. A new tool wear condition monitoring method based on deep learning under small samples. *Measurement*, 189:110622, 2022.
- [35] Yang Xu, Zhixiong Li, Shuqing Wang, Weihua Li, Thompson Sarkodie-Gyan, and Shizhe Feng. A hybrid deep-learning model for fault diagnosis of rolling bearings. *Measurement*, 169:108502, 2021.
- [36] Rüstem Binali, Havva Demirpolat, Mustafa Kuntoğlu, Mayur Makhesana, Saeed Yaghoubi, and Bahar Sayın Kul. A comprehensive review on low-cost mems accelerometers for vibration measurement: Types, novel designs, performance evaluation, and applications. *Journal of Molecular and Engineering Materials*, 12(03):2430002, 2024.

- [37] Waqas Amin Gill, Ian Howard, Ilyas Mazhar, and Kristoffer McKee. A review of mems vibrating gyroscopes and their reliability issues in harsh environments. *Sensors*, 22(19), 2022.
- [38] Adam Jabłoński, Michał Żegleń, Wojciech Staszewski, Piotr Czop, and Tomasz Barszcz. How to build a vibration monitoring system on your own? In Anna Timofiejczuk, Fakher Chaari, Radoslaw Zimroz, Walter Bartelmus, and Mohamed Haddar, editors, *Advances in Condition Monitoring of Machinery in Non-Stationary Operations*, pages 111–121, Cham, 2018. Springer International Publishing.
- [39] Marcus Varanis, Anderson Silva, Arthur Mereles, and Robson Pederiva. MemS accelerometers for mechanical vibrations analysis: a comprehensive review with applications. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 40:1–18, 2018.
- [40] Alessandro Sabato, Christopher Niezrecki, and Giancarlo Fortino. Wireless mems-based accelerometer sensor boards for structural vibration monitoring: A review. *IEEE Sensors Journal*, 17(2):226–235, 2017.
- [41] Prashanth Ragam and Nimaje Devidas Sahebraoji. Application of mems-based accelerometer wireless sensor systems for monitoring of blast-induced ground vibration and structural health: a review. *IET Wireless Sensor Systems*, 9(3):103–109, 2019.
- [42] Shipeng Han, Zhen Meng, Olatunji Omisore, Toluwanimi Akinyemi, and Yuepeng Yan. Random error reduction algorithms for mems inertial sensor accuracy improvement—a review. *Micromachines*, 11(11), 2020.
- [43] Shuzheng Shi, Wenping Geng, Kaixi Bi, Yikun Shi, Fen Li, Jian He, and Xiujuan Chou. High sensitivity mems accelerometer using pzt-based four l-shaped beam structure. *IEEE Sensors Journal*, 22(8):7627–7636, 2022.
- [44] Alice Lanniel, Tobias Boeser, Thomas Alpert, and Maurits Ortmanns. Low-noise readout circuit for an automotive mems accelerometer. *IEEE Open Journal of the Solid-State Circuits Society*, 1:140–148, 2021.
- [45] Xiang Xu, Shuang Wu, Weidong Fang, Zhe Yu, Zeyu Jia, Xiaoxu Wang, Jian Bai, and Qianbo Lu. Bandwidth optimization of mems accelerometers in fluid medium environment. *Sensors*, 22(24), 2022.
- [46] Cenk Acar and Andrei M Shkel. Experimental evaluation and comparative analysis of commercial variable-capacitance mems accelerometers. *Journal of Micromechanics and Microengineering*, 13(5):634, may 2003.

- [47] Jong-Duk Son, Byung-Hyun Ahn, Jeong-Min Ha, and Byeong-Keun Choi. An availability of mems-based accelerometers and current sensors in machinery fault diagnosis. *Measurement*, 94:680–691, 2016.
- [48] Dileep Kumar, Sanaullah Mehran Ujjan, Kapal Dev, Sunder Ali Khowaja, Naveed Anwar Bhatti, and Tanweer Hussain. Towards soft real-time fault diagnosis for edge devices in industrial iot using deep domain adaptation training strategy. *Journal of Parallel and Distributed Computing*, 160:90–99, 2022.
- [49] Sheikh Ferdoush and Xinrong Li. Wireless sensor network system design using raspberry pi and arduino for environmental monitoring applications. *Procedia Computer Science*, 34:103–110, 2014. The 9th International Conference on Future Networks and Communications (FNC’14)/The 11th International Conference on Mobile Systems and Pervasive Computing (MobiSPC’14)/Affiliated Workshops.
- [50] Billel Bengherbia, Mohamed Ould Zmirli, Abdelmoghni Toubal, and Abderrezak Guessoum. Fpga-based wireless sensor nodes for vibration monitoring system and fault diagnosis. *Measurement*, 101:81–92, 2017.
- [51] Juan Jose Saucedo-Dorantes, Miguel Delgado-Prieto, Juan Antonio Ortega-Redondo, Roque Alfredo Osornio-Rios, and Rene de Jesus Romero-Troncoso. Multiple-fault detection methodology based on vibration and current analysis applied to bearings in induction motors and gearboxes on the kinematic chain. *Shock and Vibration*, 2016(1):5467643, 2016.
- [52] Joel Igba, Kazem Alemzadeh, Christopher Durugbo, and Egill Thor Eiriksson. Analysing rms and peak values of vibration signals for condition monitoring of wind turbine gearboxes. *Renewable Energy*, 91:90–106, 2016.
- [53] Manish Vishwakarma, Rajesh Purohit, V. Harshlata, and P. Rajput. Vibration analysis and condition monitoring for rotating machines: A review. *Materials Today: Proceedings*, 4(2, Part A):2659–2664, 2017. 5th International Conference of Materials Processing and Characterization (ICMPC 2016).
- [54] WALTER BARTELMUS, FAKHER CHAARI, Radosław Zimroz, and Mohamed Haddar. Modelling of gearbox dynamics under time-varying nonstationary load for distributed fault detection and diagnosis. *European Journal of Mechanics, A/Solids*, 29(4):637–646, jul 2010.
- [55] Jeremy Sheldon, Genna Mott, Hyungdae Lee, and Matthew Watson. Robust wind turbine gearbox fault detection. *Wind Energy*, 17(5):745–755, 2014.

- [56] A. Krishnakumari, A. Elayaperumal, M. Saravanan, and C. Arvindan. Fault diagnostics of spur gear using decision tree and fuzzy classifier. *The International Journal of Advanced Manufacturing Technology*, 89(9):3487–3494, 2017.
- [57] Amit Shrivastava and Sulochana Wadhvani. An approach for fault detection and diagnosis of rotating electrical machine using vibration signal analysis. In *International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014)*, pages 1–6, 2014.
- [58] N. Aiswarya, S. Suja Priyadharsini, and K. S. Moni. An efficient approach for the diagnosis of faults in turbo pump of liquid rocket engine by employing fft and time-domain features. *Australian Journal of Mechanical Engineering*, 16(3):163–172, 2018.
- [59] Sandeep S. Udmale and Sanjay Kumar Singh. Application of spectral kurtosis and improved extreme learning machine for bearing fault classification. *IEEE Transactions on Instrumentation and Measurement*, 68(11):4222–4233, 2019.
- [60] Jérôme Antoni. Fast computation of the kurtogram for the detection of transient faults. *Mechanical Systems and Signal Processing*, 21(1):108–124, 2007.
- [61] Sheng Fu, Kun Liu, Yonggang Xu, and Yi Liu. Rolling bearing diagnosing method based on time domain analysis and adaptive fuzzy c-means clustering. *Shock and Vibration*, 2016(1):9412787, 2016.
- [62] Runesson, Julia. Vibration analysis for condition monitoring of mechanical presses, 2019. Student Paper.
- [63] L. Hsiung-Cheng and Y. Yu-Chen. Reviews of bearing vibration measurement using fast fourier transform and enhanced fast fourier transform algorithms. *Advances in Mechanical Engineering*, 11(1):1–12, 2019.
- [64] R. B. Randall. *Frequency Analysis*. Bruel and Kjaer, Copenhagen, Denmark, 1987.
- [65] D. Abboud, J. Antoni, S. Sieg-Zieba, and M. Eltabach. Envelope analysis of rotating machine vibrations in variable speed conditions: A comprehensive treatment. *Mechanical Systems and Signal Processing*, 84:200–226, 2017.
- [66] D. HO and R.B. RANDALL. Optimisation of bearing diagnostic techniques using simulated and actual bearing fault signals. *Mechanical Systems and Signal Processing*, 14(5):763–788, 2000.

- [67] R. RUBINI and U. MENEGHETTI. Application of the envelope and wavelet transform analyses for the diagnosis of incipient faults in ball bearings. *Mechanical Systems and Signal Processing*, 15(2):287–302, 2001.
- [68] Achmad Widodo, Eric Y. Kim, Jong-Duk Son, Bo-Suk Yang, Andy C.C. Tan, Dong-Sik Gu, Byeong-Keun Choi, and Joseph Mathew. Fault diagnosis of low speed bearing based on relevance vector machine and support vector machine. *Expert Systems with Applications*, 36(3, Part 2):7252–7261, 2009.
- [69] Valéria C. M. N. Leite, Jonas Guedes Borges da Silva, Giscard Francimeire Cintra Veloso, Luiz Eduardo Borges da Silva, Germano Lambert-Torres, Erik Leandro Bonaldi, and Levy Ely de Lacerda de Oliveira. Detection of localized bearing faults in induction machines by spectral kurtosis and envelope analysis of stator current. *IEEE Transactions on Industrial Electronics*, 62(3):1855–1865, 2015.
- [70] M. Ahsan and D. Bismor. Early-stage fault diagnosis for rotating element bearing using improved harmony search algorithm with different fitness functions. *IEEE Transactions on Instrumentation and Measurement*, 71:1–9, 2022.
- [71] Lucio Ciabattoni, Francesco Ferracuti, Alessandro Freddi, and Andrea Monteriù. Statistical spectral analysis for fault diagnosis of rotating machines. *IEEE Transactions on Industrial Electronics*, 65(5):4301–4310, 2018.
- [72] Yu Wei, Yuqing Li, Minqiang Xu, and Wenhua Huang. A review of early fault diagnosis approaches and their applications in rotating machinery. *Entropy*, 21(4), 2019.
- [73] Zhipeng Feng, Ming Liang, and Fulei Chu. Recent advances in time–frequency analysis methods for machinery fault diagnosis: A review with application examples. *Mechanical Systems and Signal Processing*, 38(1):165–205, 2013. Condition monitoring of machines in non-stationary operations.
- [74] J Zou and J Chen. A comparative study on time–frequency feature of cracked rotor by wigner–ville distribution and wavelet transform. *Journal of Sound and Vibration*, 276(1):1–11, 2004.
- [75] Manel Rhif, Ali Ben Abbes, Imed Riadh Farah, Beatriz Martínez, and Yanfang Sang. Wavelet transform application for/in non-stationary time-series analysis: A review. *Applied Sciences*, 9(7), 2019.
- [76] D. Goyal and B. S. Pabla. The vibration monitoring methods and signal processing techniques for structural health monitoring: A review. *Archives of Computational Methods in Engineering*, 23(4):585–594, 2016.

- [77] Shengxiang Huang, Xinpeng Wang, Chenfeng Li, and Chao Kang. Data decomposition method combining permutation entropy and spectral substitution with ensemble empirical mode decomposition. *Measurement*, 139:438–453, 2019.
- [78] Z.K. Peng, Peter W. Tse, and F.L. Chu. A comparison study of improved hilbert–huang transform and wavelet transform: Application to fault diagnosis for rolling bearing. *Mechanical Systems and Signal Processing*, 19(5):974–988, 2005.
- [79] T. Y. Wu, Y. L. Chung, and C. H. Liu. Looseness diagnosis of rotating machinery via vibration analysis through hilbert–huang transform approach. *Journal of Vibration and Acoustics*, 132(3):031005, 04 2010.
- [80] Shazali Osman and Wilson Wang. A normalized hilbert-huang transform technique for bearing fault detection. *Journal of Vibration and Control*, 22(11):2771–2787, 2016.
- [81] D. Gabor. Theory of communication. part 1: The analysis of information. *Journal of the Institution of Electrical Engineers - Part III: Radio and Communication Engineering*, 93:429–441, 1946.
- [82] Jérôme Antoni. The spectral kurtosis: a useful tool for characterising non-stationary signals. *Mechanical Systems and Signal Processing*, 20(2):282–307, 2006.
- [83] Fabien Millioz and Nadine Martin. Circularity of the stft and spectral kurtosis for time-frequency segmentation in gaussian environment. *IEEE Transactions on Signal Processing*, 59(2):515–524, 2011.
- [84] Jordi Burriel-Valencia, Ruben Puche-Panadero, Javier Martinez-Roman, Angel Sapena-Bano, and Manuel Pineda-Sanchez. Short-frequency fourier transform for fault diagnosis of induction machines working in transient regime. *IEEE Transactions on Instrumentation and Measurement*, 66(3):432–440, 2017.
- [85] Demetgul Mustafa, Zhao Yicheng, Gu Minjie, Hillenbrand Jonas, and Fleischer Jürgen. Motor current based misalignment diagnosis on linear axes with short-time fourier transform (stft). *Procedia CIRP*, 106:239–243, 2022. 9th CIRP Conference on Assembly Technology and Systems.
- [86] Razan Issa, Guy Clerc, Malorie Hologne-Carpentier, Ryan Michaud, Eric Lorca, Christophe Magnette, and Anes Messadi. Review of fault diagnosis methods for induction machines in railway traction applications. *Energies*, 17(11), 2024.

- [87] Dongdong Liu, Weidong Cheng, and Weigang Wen. Rolling bearing fault diagnosis via stft and improved instantaneous frequency estimation method. *Procedia Manufacturing*, 49:166–172, 2020. Proceedings of the 8th International Conference on Through-Life Engineering Services – TESConf 2019.
- [88] Cai Yi, Hao Wang, Le Ran, Lu Zhou, and Jianhui Lin. Power spectral density-guided variational mode decomposition for the compound fault diagnosis of rolling bearings. *Measurement*, 199:111494, 2022.
- [89] BeagleBoard.org Foundation. Beaglebone black. <https://docs.beagleboard.org/boards/beaglebone/black/index.html>. Accessed: 2024-11-16.
- [90] Muhammad Ahsan and Dariusz Bismor. Calibration of a high sampling frequency mems-based vibration measurement system. In Marek Pawelczyk, Dariusz Bismor, Szymon Ogonowski, and Janusz Kacprzyk, editors, *Advanced, Contemporary Control*, pages 294–302, Cham, 2023. Springer Nature Switzerland.
- [91] Chih-Yung Huang and Jian-Hao Chen. Development of dual-axis mems accelerometers for machine tools vibration monitoring. *Applied Sciences*, 6(7), 2016.
- [92] D. Buchczik and M. Pawelczyk. Calibration and evaluation of accelerometers based on multisinusoidal excitation. *Diagnostyka*, 18(4):79–87, 2017.
- [93] J. Lee, H. Qiu, G. Yu, J. Lin, and Rexnord Technical Services (2007). Ims, university of cincinnati, bearing data set. <http://ti.arc.nasa.gov/project/prognostic-data-repository>. Accessed: 2021-03-10.
- [94] Vinayak Tyagi. Bearing dataset. <https://www.kaggle.com/datasets/vinayak123tyagi/bearing-dataset/data>, 2020. Accessed: 2021-04-01.
- [95] MFPT (The Society for Machinery Failure Prevention Technology). Fault data sets, n.d. Accessed: 2021-03-12.
- [96] Kamal Jafarian, Mohammadsadegh Mobin, Ruholla Jafari-Marandi, and Elaheh Rabiei. Misfire and valve clearance faults detection in the combustion engines based on a multi-sensor vibration signal monitoring. *Measurement*, 128:527–536, 2018.
- [97] Cheng Gu, Xin-Yong Qiao, Huaying Li, and Ying Jin. Misfire fault diagnosis method for diesel engine based on memd and dispersion entropy. *Shock and Vibration*, 2021(1):9213697, 2021.
- [98] Nayara Formiga Rodrigues, Alisson V. Brito, Jorge Gabriel Gomes Souza Ramos, Koje Daniel Vasconcelos Mishina, Francisco Antonio Belo, and Abel Cavalcante

- Lima Filho. Misfire detection in automotive engines using a smartphone through wavelet and chaos analysis. *Sensors*, 22(14), 2022.
- [99] Mohd Khairul Nizam bin Md Sharib, Elmi bin Abu Bakar, Ahmad Faizul bin Hawary, Mohammad Nishat Akhtar, and Mohamad Nazir Abdullah. Adaptive system on engine misfire and knocking features for ic engine. *Materials Today: Proceedings*, 2023.
- [100] M. Ahsan, D. Bismor, and P. FABIŚ. Analysis of the vehicle engine misfires using frequency-domain approaches at various rpms with adxl1002 accelerometer. *Archives of Acoustics*, pages 1–10, 2024.
- [101] D. Bismor. Analysis and comparison of vibration signals from internal combustion engine acquired using piezoelectric and mems accelerometers. *Vibration in Physical Systems*, 30(1,2019112):1–8, 2019.
- [102] M. Ahsan, D. Bismor, and M. Arslan. Arl-wavelet-bpf optimization using pso algorithm for bearing fault diagnosis. *Archives of Control Sciences*, 33(3):589–606, 2023.
- [103] Yuxiong Li, Xianzhen Huang, Pengfei Ding, and Chengying Zhao. Wiener-based remaining useful life prediction of rolling bearings using improved kalman filtering and adaptive modification. *Measurement*, 182:109706, 2021.
- [104] Cédric Peeters, Jérôme Antoni, and Jan Helsen. Blind filters based on envelope spectrum sparsity indicators for bearing and gear vibration-based condition monitoring. *Mechanical Systems and Signal Processing*, 138:106556, 2020.
- [105] Z. W. Geem, J. H. Kim, and G. V. Loganathan. A new heuristic optimization algorithm: Harmony search. *Simulation*, 76(2):60–68, 2001. Cited By :4224.
- [106] Kang Seok Lee, Zong Woo Geem, Sang ho Lee, and Kyu woong Bae. The harmony search heuristic algorithm for discrete structural optimization. *Engineering Optimization*, 37(7):663–684, 2005.
- [107] Kaizhou Gao, Zhiguang Cao, Le Zhang, Zhenghua Chen, Yuyan Han, and Quanke Pan. A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems. *IEEE/CAA Journal of Automatica Sinica*, 6(4):904–916, 2019.
- [108] Xuewen Xia, Ying Xing, Bo Wei, Yinglong Zhang, Xiong Li, Xianli Deng, and Ling Gui. A fitness-based multi-role particle swarm optimization. *Swarm and Evolutionary Computation*, 44:349–364, 2019.

- [109] Xiaoxia Yu, Baoping Tang, and Kai Zhang. Fault diagnosis of wind turbine gearbox using a novel method of fast deep graph convolutional networks. *IEEE Transactions on Instrumentation and Measurement*, 70:1–14, 2021.
- [110] Mohammed Hakim, Abdoulhdi A. Borhana Omran, Ali Najah Ahmed, Muhanad Al-Waily, and Abdallah Abdellatif. A systematic review of rolling bearing fault diagnoses based on deep learning and transfer learning: Taxonomy, overview, application, open challenges, weaknesses and recommendations. *Ain Shams Engineering Journal*, 14(4):101945, 2023.
- [111] Muhammad Ahsan and Mostafa M. Salah. Similarity index of the stft-based health diagnosis of variable speed rotating machines. *Intelligent Systems with Applications*, 20:200270, 2023.
- [112] Hao Zhang, Teng Li, Ai Jing, and Siyuan Yang. Sequence–spectrogram fusion network for wind turbine diagnosis through few-shot time-series classification. *Advanced Engineering Informatics*, 64:102976, 2025.
- [113] Mona Yadi, Yoshiharu Morimoto, and Yasuhiro Takaya. Vibration distribution measurement of car door and engine head using oppa vibration distribution analyzer. *Vibration*, 6(2):421–433, 2023.
- [114] Jian Chen, Robert Bond Randall, and Bart Peeters. Advanced diagnostic system for piston slap faults in ic engines, based on the non-stationary characteristics of the vibration signals. *Mechanical Systems and Signal Processing*, 75:434–454, 2016.
- [115] Xiaohui Duan and Zhipeng Feng. Time-varying filtering for nonstationary signal analysis of rotating machinery: Principle and applications. *Mechanical Systems and Signal Processing*, 192:110204, 2023.
- [116] Ihor Javorskyj, Roman Yuzefovych, Oleh Lychak, George Trokhym, and Mykola Varyvoda. Methods of periodically non-stationary random processes for vibrations monitoring of rolling bearing with damaged outer race. *Digital Signal Processing*, 145:104343, 2024.
- [117] Marcus Varanis, Anderson L. Silva, José M. Balthazar, and Robson Pederiva. A tutorial review on time-frequency analysis of non-stationary vibration signals with nonlinear dynamics applications. *Brazilian Journal of Physics*, 51:859–877, 2021.
- [118] Bayu Adhi Tama, Malinda Vania, Seungchul Lee, and Sunghoon Lim. Recent advances in the application of deep learning for fault diagnosis of rotating machinery using vibration signals. *Artificial Intelligence Review*, 56:4467–4709, 2023.

- [119] Omri Matania, Itai Dattner, Jacob Bortman, Ron S. Kenett, and Yisrael Parmet. A systematic literature review of deep learning for vibration-based fault diagnosis of critical rotating machinery: Limitations and challenges. *Journal of Sound and Vibration*, 590:118562, 2024.
- [120] Muhammad Ahsan, Dariusz Bismor, and Paweł Fabiś. Comparison of ann and cnn models for misfire detection in vehicle engine at different rpms with low-cost adxl1002 accelerometer. In *2024 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, pages 60–65, 2024.
- [121] Syed Muhammad Tayyab, Steven Chatterton, and Paolo Pennacchi. Intelligent defect diagnosis of rolling element bearings under variable operating conditions using convolutional neural network and order maps. *Sensors*, 22(5), 2022.
- [122] Pauline Ong, Yean Keong Tan, Kee Huong Lai, and Chee Kiong Sia. A deep convolutional neural network for vibration-based health-monitoring of rotating machinery. *Decision Analytics Journal*, 7:100219, 2023.
- [123] Muhammad Ahsan and Mostafa M. Salah. Efficient dcnn-lstm model for fault diagnosis of raw vibration signals: Applications to variable speed rotating machines and diverse fault depths datasets. *Symmetry*, 15(7), 2023.
- [124] Muhammad Ahsan and Dariusz Bismor. Advanced fault diagnosis in rotating machines using 2d grayscale images with improved deep convolutional neural networks. In *2023 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, pages 77–82, 2023.
- [125] A. Zeiler, R. Faltermeier, I. R. Keck, A. M. Tomé, C. G. Puntonet, and E. W. Lang. Empirical mode decomposition - an introduction. In *The 2010 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2010.
- [126] Ayan Kumar Dutta, Brejesh Lall, and Shiv Dutt Joshi. Empirical mode decomposition techniques: A simulated review. In *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pages 1–5, 2023.