



An IoT-Based Monitoring System for Induction Motor Faults Utilizing Deep Learning Models

Eyüp Irgat¹ · Eyüp Çınar^{3,4} · Abdurrahman Ünsal² · Ahmet Yazıcı^{3,4}

Received: 16 May 2022 / Revised: 23 September 2022 / Accepted: 26 October 2022 / Published online: 5 November 2022
© Krishtel eMaging Solutions Private Limited 2022

Abstract

Purpose Electrical motors are among the most widely used equipment components across many industries. Therefore, monitoring electrical motors for the early detection of faults is essential for uninterrupted production and can save time and money for manufacturers. Bearing faults are one of the most frequently encountered fault types in induction motors. Although standard offline datasets and algorithm designs have been studied extensively in recent years, they lack a full-scale IoT-based monitoring system design for data collection and deployment in the field. In this paper, outer race, inner race, and ball faults of the bearings of a three-phase (1.5 kW, 6 poles) induction motor are studied using vibration signals and an IoT-based monitoring system.

Methods The vibration signals are collected on a new testbed where the Edge processing units can process the signals and also transfer the sensor data over an IoT-based system. Raw data signals, such as vibration, current, and torque, are preprocessed at the Edge. Features are transferred to a database with Message Queuing Telemetry Transport (MQTT) for long-term storage. Only the vibration signals are analyzed for the detection of bearing faults. The monitoring of vibration signals of the motor can be implemented online or offline locally at the Edge with the implemented IoT system. A Convolution Neural Network (CNN)-based deep learning algorithm is utilized to establish a data-driven condition monitoring AI model. The vibration signals are converted into spectrograms using Short Time Fourier Transform (STFT), and then the CNN model is trained.

Results The proposed IoT-based monitoring system solution, combined with the AI method, can successfully process the sensor signals from the motor. The model can distinguish a healthy bearing from three types of faulty bearings with an average accuracy rate of 95%.

Conclusion In this study, unlike the majority of the studies in the literature focusing on algorithm development with standard datasets for motor faults, a new induction motor condition monitoring system based on IoT technologies is designed and implemented. The system is scalable and can be deployed to monitor a fleet of motor equipment in manufacturing. Based on the needs, the AI model can run both at the Edge and at the server side with data analytics tools to monitor the condition of induction motor faults.

Keywords Induction motor · Vibration signal · Bearing fault · IoT · CNN

Introduction

Internet of Things (IoT) technology has been used in many applications, including production lines, industrial applications, energy, telecommunication applications, the automotive industry, and condition monitoring and fault diagnosis of electrical machines. Manufacturing industries are

leveraging this technology to improve the accuracy and quality of the facility's existing monitoring and control system [6]. IoT, a networked system of physical objects, is used to write data to a database over the Internet or local networks. IoT provides remote sensing and control of things over an existing network infrastructure, creating opportunities to integrate the physical world into computer-based systems directly. Monitoring, detecting, and diagnosing faults of induction motors are essential for all applications, especially in the early state. Early detection, diagnostics, and the sign of motor faults allow appropriately timed servicing

✉ Eyüp Irgat
eyup.irgat@dpu.edu.tr

Extended author information available on the last page of the article

to avoid expensive motor failures, avoid costly economic losses related to process interruption, prevent unexpected interruption, and enable maximum production [9].

IoT devices communicate with each other using protocols. MQTT is the most widely used protocol for IoT applications [17]. It is used as an interface layer between users and devices. It is an intermediate layer standard for asynchronous message transmission based on TCP/IP. It is designed as a light broadcast/subscription messaging protocol, ideal for connecting remote devices with less code and minimum network bandwidth [4]. The MQTT protocol system has lower power consumption and a higher sending rate. MQTT is a widely preferred and used protocol due to its higher quality of service [17]. Since the client in an MQTT protocol does not require a request update which saves bandwidth, the Publish/Subscribe is more suitable and preferred for IoT applications. An application of the monitoring system, which writes the data to a database with MQTT protocol, is given in [13]. The MQTT uses Publish/Subscribe or Request/Response architecture.

Electrical machines are used in many applications, including developing and automating modern industrial applications. Induction motors are among the most used electrical machines. Their faults can lead to undesirable consequences like production line fault, process interruption, high maintenance costs, and safety hazards. Therefore, maintaining induction motors' good physical condition is essential for most industries. In most practices, induction motors are used in severe conditions and are subject to several faults because of thermal, mechanical, and other ambient stress conditions [10]. The induction motor faults can be categorized as mechanical faults and electrical faults. Mechanical faults contain misalignment, air gap eccentricity, and bearing faults, while electrical faults of induction motors involve rotor and stator faults.

It has been reported in [10] that 41% of the total faults of induction motors are due to bearing faults, 36% to stator faults, 9% to rotor faults, and 14% to other motor faults. In [19], the bearing faults are responsible for approximately 40% and 50% of the total faults of induction motors. In [25], the bearing faults accounted for 30% and 40% of motor faults. Bearings are the most significant parts of induction motors. An unexpected fault in a bearing may cause the motor's corruption, damage, and consequently economic losses [22]. Bearing faults are accounted for 40% of breakdowns in large rotating machinery systems and 90% in small rotating machinery systems [9]. Bearing faults are among the most common reasons for electrical machine faults [20]. Therefore, this study focuses on bearing faults.

Industrial maintenance practice has been changed parallel to the improvement of technology. The earliest form of maintenance is breakdown maintenance, where no actions are taken for maintenance until the equipment breaks down

and, as a result, needs a repair or replacement. Then preventive maintenance was used, which requires maintenance for a period notwithstanding the equipment's physical condition. In recent years, predictive maintenance programs have been used [5]. Predictive maintenance is a condition-based preventive maintenance program comprising physical condition rating, detection, and lifetime prognostics. Therefore, this study contributes to predictive maintenance by assessing the health of bearings of induction motors. The implemented IoT data pipeline and long-term historical data storage can also be expanded to prognostic analytics and models.

The detection, diagnostics, and prognostics studies may detect and isolate faults, predict the future health condition, estimate the remaining useful life (RUL) of induction motors, and prevent performance deterioration, malfunction, and sudden failures [23].

In this study, an implemented IoT-based system is used to monitor the condition of the bearings of a three-phase induction motor. Bearing faults can be monitored and detected remotely using IoT technologies. The monitoring system enables sensor data processing and detecting bearing faults both locally at the Edge and remotely. This study selected vibration signals to process and detect bearing faults. The data pipeline can help transfer raw data and preprocessed data at the Edge. The proposed IoT-based system can be used to monitor multiple induction motors in real time by recording, processing, and transferring vibration signals.

The single-row bearings with two rings, specifically the inner and outer rings, are utilized in induction motors. A set of balls rotate between the inner and outer ring, as illustrated in Fig. 1. The bearing faults may be caused by attrition, tear, aging, overloads, imbalances, and overheating [11]. Bearing faults appear in the inner race, outer race, and ball. Each bearing fault has its characteristic frequency in the vibration and stator current signals of the induction motor depending on the construction and mechanical dimensions of the bearing.

The fault diagnosis methods of induction motor faults can be categorized as model-based, signature-extraction-based, and knowledge-based. The model-based methods use

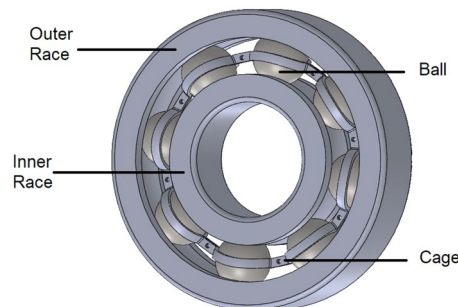


Fig. 1 Components of a single-row bearing

a mathematical model of induction motors to detect faults. In the signature-extraction-based approaches, specific fault signatures are extracted from the monitored signals, such as motor drive current. The knowledge-based methods use signal processing and machine learning (ML) techniques to build a data-driven model instead of a specific mathematical model of an induction motor [2, 3].

Signal processing methods can be implemented in the frequency domain, time domain, and time–frequency domain [8]. The Wavelet Transformation (WT), Empirical Mode Decomposition (EMD), Fast Fourier Transform (FFT) [21], Wavelet Packet Transform (WPT), and Discrete Wavelet Transform [15] are the widely used methods for the processing of vibration signals both in time domain and frequency.

In the time domain analysis, statistical features like peak-to-peak value, root-mean-square, skewness, crest factor, mean, and kurtosis are used to determine the condition of bearings [7]. In the frequency domain analysis, monitoring signals are transferred from the time domain to the frequency domain using FFT and discrete Fourier transform (DFT). The characteristic frequencies are easily detected in the frequency domain. The frequency domain analysis is restricted to non-stationary signals and can be more effective in analyzing stationary signals. Since the vibration signals (also used in this study) are non-stationary, time–frequency analysis methods are preferably used. The commonly used transformation methods for time–frequency analysis include Wavelet Transform and Short Time Fourier Transform (STFT) [8]. If the peak amplitudes of the vibrations signal are not high, FFT-based methods are ineffective. Selection of the proper mother wavelet in a Wavelet Transform, the selection of separation level, and frequency band require professional knowledge.

Another challenge in fault detection is determining appropriate rule-based alarm conditions from features after time and frequency domain analysis. This is since the equipment baseline drifts over time. ML-based methods are an alternative solution to overcome this limitation by building a more sophisticated data-driven model that can generate the rules. ML-based classification algorithms are used in a wide range of studies, including determining faults of electrical machines with the development of high-performance/speed computers [2, 3]. ML algorithms can be classified into supervised ML and unsupervised ML. Classification is one of the supervised ML tasks. An ML model, once trained, works in three stages [14]: It takes the data, learns patterns in the data, and predicts the patterns for the unseen data. Artificial Neural Networks (ANN) [1], Decision Tree (DT), k -Nearest Neighbors (k -NN), and Support Vector Machines (SVM) are among the frequently encountered supervised ML algorithm examples [24]. A deep learning algorithm is an ML algorithm that uses Artificial Neural Networks with one or more hidden layers. In most ML methods, data

preprocessing is required, and features are extracted from the raw data and given as input to the algorithm. However, deep learning algorithms extract the features themselves. Generally having a higher model capacity, deep learning methods can model complex operating conditions and give accurate predictions [18].

A Convolutional Neural Network (CNN) is a deep learning algorithm that can extract various features from an input image by convolution operations and create a powerful image recognition model. Although commonly used for image pattern recognition tasks, CNNs and their variations gain attention in time series analysis and parallel to diagnose equipment faults [16]. They can learn hierarchical representations from the input images via multiple hidden layers [9, 18, 20]. CNN can successfully obtain spatial attributes in an image due to the application of the corresponding filters. The network can be trained to understand the complex pattern perfectly. This can be successfully adapted for recognizing time–frequency patterns from a spectrogram obtained by STFT operations.

The main contributions of this study can be listed as follows:

- An end-to-end motor monitoring system was designed by LabVIEW at the Edge and IoT technologies, including a new middleware platform for this use case for the first time.
- Using deep learning algorithms, standalone processing was applied by importing LabVIEW modules on a PC and on the CompactRio (Edge).
- A robust data pipeline was established to record and process vibration signals of an induction motor for condition monitoring by extracting characteristic features for diagnosing bearing faults in the short term and long term by analyzing the recorded data historically later.

The remainder of this paper is organized as follows. The proposed method and experimental setup are given in Sect. 2. The experimental results are given in Sect. 3, and finally, conclusions are presented in Sect. 4.

The Proposed Method

Each bearing fault is artificially created by drilling holes with a diameter of 2.5 mm in the outer race, inner race, and bearing ball using the electroplating method. Every bearing fault is studied independently. Figure 2 shows the implemented bearing faults.

The induction motor was tested under four cases: healthy bearing, outer-race faulty bearing, inner-race faulty bearing, and faulty ball bearing. The vibration signals of the induction motor with healthy and faulty bearings were recorded

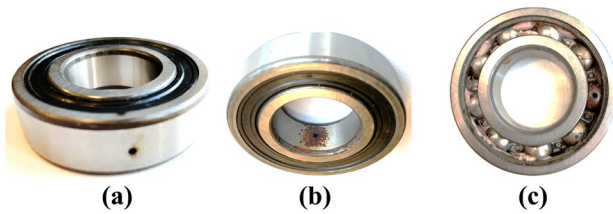


Fig. 2 Bearing fault classes including **a** outer race, **b** inner race, and **c** ball defects (left to right)

with a sampling frequency of 6400 Hz and a duration of 40 s. The recorded vibration signals were converted into spectrograms using Short Time Fourier Transform (STFT) and then used to train the proposed CNN model. The trained model is used to detect the faulty bearings in real time.

Experimental Setup

A schematic diagram representing the testbed design is given in Fig. 3. The experimental setup consists of a 1.5 kW 6-poles induction motor loaded with an electromagnetic powder brake (Merobel Frat 650). The brake is controlled with current signals through a brake control circuit and can generate linear torque values up to 65Nm. A snail fan blower housing is designed and placed around the brake to remove excess heat generated by the brake.

A National Instrument (NI) data acquisition system (CompactRio with 8-Slot, 1.33 GHz CPU, 1 GB Ram, and 4 GB Hard disk) is used for data acquisition in the testbed. A three-axis accelerometer sensor (PCB triaxial Model 356A15) measures the vibration through a NI 9230 module. The induction motor's torque information is acquired through a torque sensor (ETH DRBK 50) and transferred into the DAQ system through a NI 9209 module.

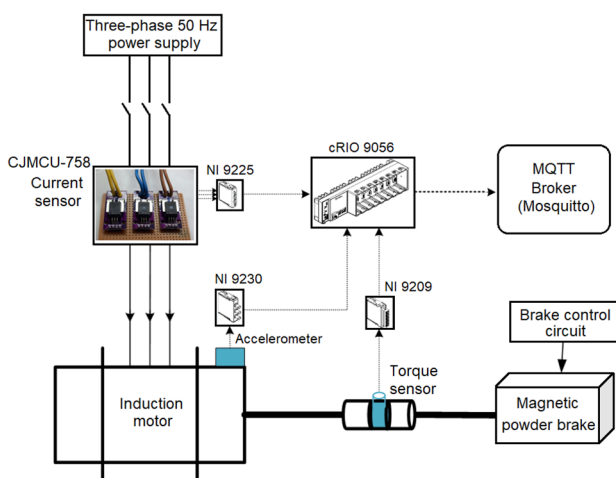


Fig. 3 Schematic diagram testbed

Figure 4 shows the realized testbed located in a lab environment. The entire testbed sits on a metal table with supporting antivibration shock observers underneath to eliminate the noise caused by the other parts. The testbed design is highly configurable by the holes created on the metal sheet to adapt to different motor test case scenarios.

The LabVIEW software tool [12] is used for signal processing. A LabVIEW interface with three modules is designed for recording and processing vibrations signals. The block diagram of the interface is given in Fig. 5.

In module 1, vibration data are read and written to a CSV file on CompactRio as a buffer for temporary storage. Module 1 also includes MQTT client code blocks enabling the edge unit, if configured to send only the raw data in these files through the data pipeline. Module 2 consists of code blocks mainly configured for edge processing. The CSV data files are processed to extract various statistical features, which can be posted through the data pipeline using the MQTT client program. Especially for signals that are necessary to collect at high sampling rates, it is more desired to send preprocessed signals over the network using the data pipeline. Module 3 consists of our DL-based fault detection models at the Edge.

The end-to-end data pipeline and the middleware system architecture implemented in this study are presented in Fig. 6. The middleware software system acts as a data distributor and forwards the data to other subscribed services. The middleware software stack mainly utilizes an open-source framework called FIWARE, and its components are reprogrammed according to our specific needs in this study. The Orion Context Broker acts as a data distributor. It holds the latest state of the virtual entities subscribed to it and sends updates to other services, such as databases with subscriptions. Many different entities can simultaneously subscribe to it and send updates, like our motor testbed entity. Orion Context Broker controls subscribed entities' context records' complete lifecycle and allows queries and updates for current subscriptions. The context broker utilizes a NoSQL database called MongoDB to hold a record of context information. The second principal component of the data pipeline is called Draco. It is an Apache NiFi-based

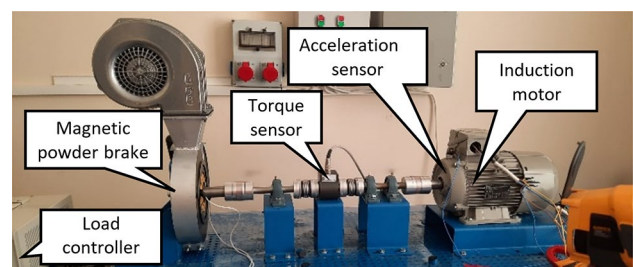


Fig. 4 Testbed

Fig. 5 LabVIEW interface modules

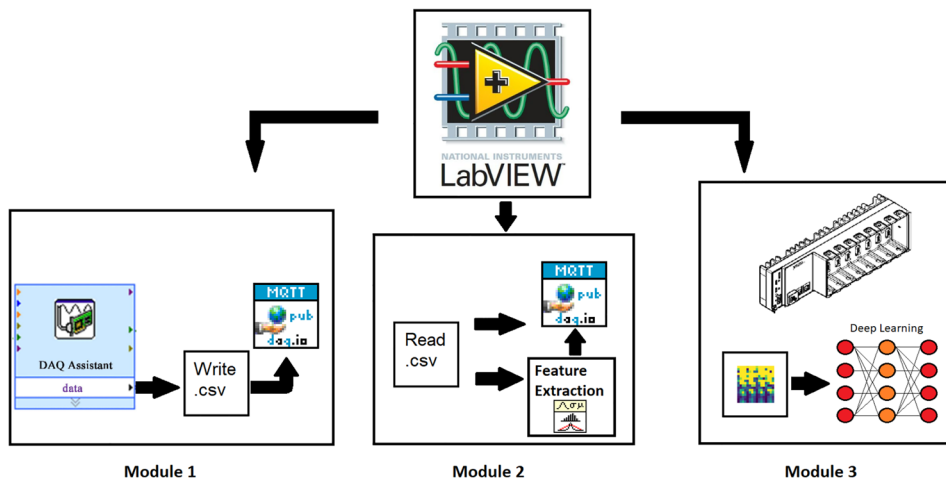
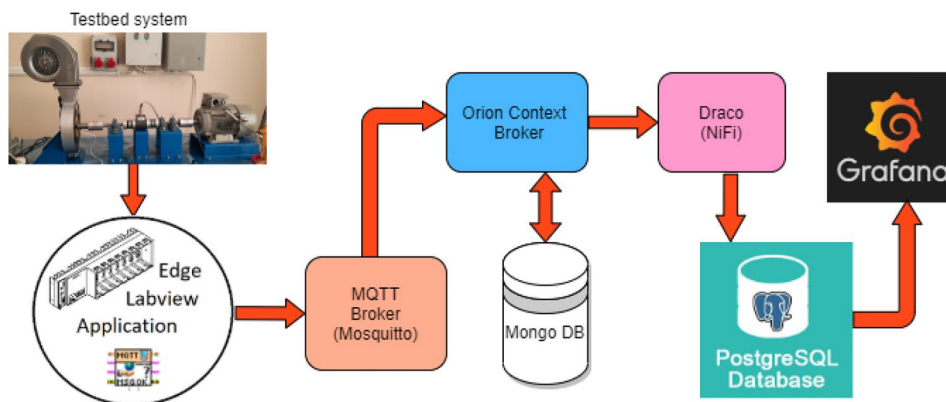


Fig. 6 IoT Middleware system



service that helps to transform data into data streams without packet loss. The data format forwarded from the context broker should be transformed into a tabular structure to be persisted into a third-party relational SQL storage called PostgreSQL. It converts context broker-specific message format to required types for a PostgreSQL columnar format. In the case of a high throughput data stream burst, Draco is configured to queue messages and process them gradually so that the data loss is prevented.

The accumulated data in the queue in Draco are transferred to the respective fields in SQL tables created in the database. At the same time, the pipeline's data transfer can be instantly monitored by any subscribed computer into the edge unit via the designed graphical user interfaces and the charts created for visualizing sensor data processing, as illustrated in the following sections.

To perform data analytics and advanced visualization of persisted data in PostgreSQL, an open-source data analytics tool called Grafana is employed. Grafana is a multi-platform open-source interactive visualization and alert-enabled web application that can be customized to visualize data. Figure 7 demonstrates an example data segment of motor vibration data's statistical features extracted at the edge unit

and transferred into the database utilizing the data pipeline. These features are *x*-vibrational axis statistical descriptors over time, including min, max, skewness, and kurtosis.

It is possible to set custom alarm threshold values based on the monitored data in the data analytics tool. For example, a specific threshold value for the max of *x*-vibration signals is set in the Grafana. If the monitored value exceeds a predetermined threshold, warning messages are sent via various channels, such as emails or mobile text messages. The platform is designed to interact with the electrical motor in two ways such that an automated halt message can be pushed down to an internet-based control relay circuitry to trigger a halt action for the motor and cut the power. Utilizing this proposed middleware system, it is possible to scale the pipeline to many induction motor fleets to monitor centrally for the health assessment of bearings. If desired, the induction motor with faulty bearings can be stopped remotely to prevent undesirable consequences. The online system can be programmed by placing business rules such that if an amplitude anomaly is detected, the edge AI model can be triggered for inference, and fault diagnosis can be performed to diagnose the anomaly from a set of predefined motor faults.

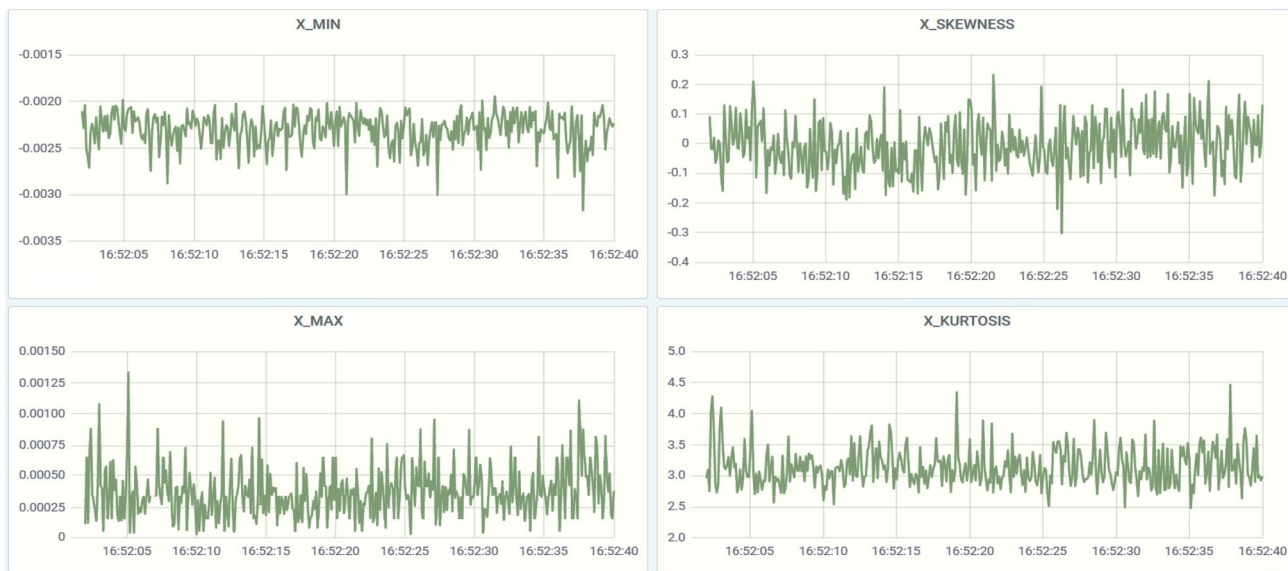


Fig. 7 An example dashboard created using the data analytics tool Grafana

Module 3 of our edge software design includes additional data preprocessing modules and the modules that can run our trained DL-based fault detection and classification models. In module 3, the raw vibration signal (offline or real time) is read and processed through a wavelet denoise filter. Then the STFT of the filtered signal is implemented to obtain spectrograms. The resulting spectrogram file is input for real-time inference of the trained CNN deep learning model at the Edge. The detection and diagnosis decisions of the bearing faults are performed in this module.

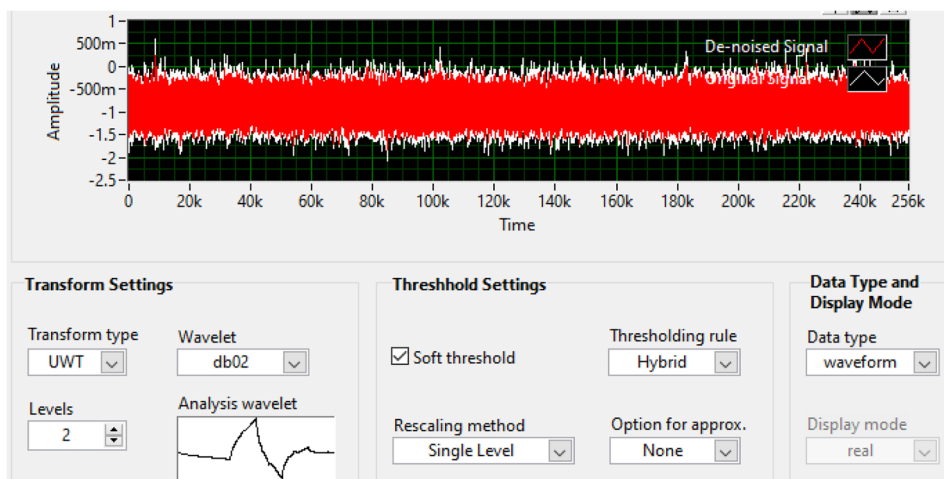
Data Preprocessing

The vibration signals at the edge unit are recorded with a sampling frequency of 6400 Hz and a length of 40 s into

CSV files. After the preprocessing steps detailed in this section are finished, the data are transferred out of the edge unit for the offline training of the DL algorithm. The experimental data collected during this phase include four classes: healthy, inner race fault, outer race fault, and ball fault. The data with a full motor load (100%) for each class are utilized in the DL algorithm training.

Wavelet denoising is a popular tool for eliminating noise from signals. This step helps filter the possible noise artifacts in the vibration and enhances data quality for the upcoming steps. The fundamental purpose is to regulate the wavelet coefficients within the new basis, and in this way, the noise may be eliminated from the data. The details of Wavelet denoise filtering parameters and the GUI designed to configure this preprocessing step are shown in Fig. 8.

Fig. 8 Wavelet denoise



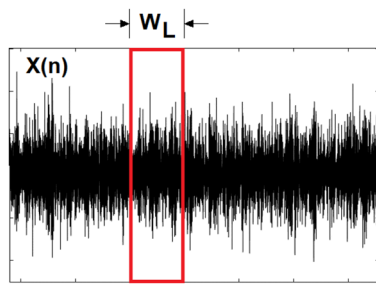


Fig. 9 A rectangular STFT window sliding over a vibrational data

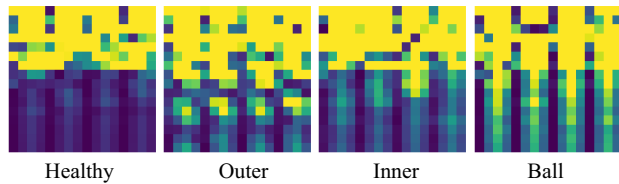


Fig. 10 Input samples of the four cases

After the raw data are passed through the denoising step, STFT transformation is applied at the Edge. As shown in a representative signal in Fig. 9, the fundamental concept of STFT is to use a fixed-size sliding window into a time-varying signal in the time domain and acquire varying frequency spectral components within this window. As a result of the transformation, a spectrogram image can be used in pattern recognition tasks with a DL algorithm such as CNN. A rectangular shape of a window having a size of 128 (W_L) samples is utilized to extract the STFT images. Figure 9 illustrates the STFT window taken over a single x -axis vibration signal. The time step between each sliding window is taken as 40 samples.

Classification Methods

A training dataset is built after the raw signal is denoised, and spectrogram images are obtained through STFT operations. Note that only x -vibrational axis data are utilized to classify fault classes in this study for fault classification tasks. The final dataset consists of 6400 samples from each fault class. Example spectrograms belonging to four classes are presented in Fig. 10.

The CNN model is trained with 70% of the training dataset, and the remaining 30% is utilized for validation. As illustrated in Fig. 11, specific CNN network parameters include three layers of varying combinations of convolution and max pooling operations with kernel sizes, as indicated in the figure. A flattening operation is applied at the end of the convolution layer, and the resulting vector is transferred through a dense layer. Since there are four classes, the output layer includes a dense layer of four neurons with a softmax activation at the end.

During the network training, Categorical_crossentropy is used as the loss function. Adam algorithm is used as the optimization algorithm, and accuracy is selected as a specific metric. The hyperparameters used for the training model are given in Table 1. The batch_size of 32 which provides the highest training and validation accuracies is used in the model. During the training, a 0.25 dropout rate is applied to avoid the overfitting of the model.

Experimental Results

This section presents the performance results of training and testing the DL model. The initial training and testing phases are first experimented on a PC connected to the edge unit and by processing the offline collected data on different days and sessions. The confusion matrixes of training and

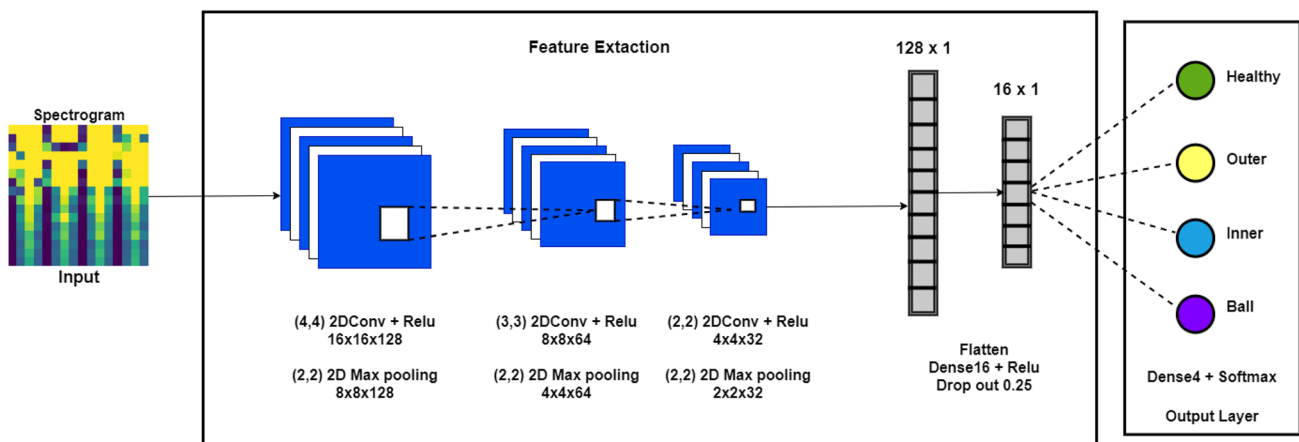


Fig. 11 Deep learning model

Table 1 Hyperparameters for training model

Hyperparameters	Value	Best fit
Optimization function	Adam	–
Learning rate	0.1–0.001	0.001
Loss function	categorical_crossentropy	–
Activation function	Relu(conv layer), softmax(output layer)	–
Dropout rate	0–0.25	0.25
Number of epochs	1–20	20
Batch_size	8, 32, 128	32

validation are given in Fig. 12. The performance of training and validation is shown in Table 2.

The training and validation accuracy of the model is given in Fig. 13. As observed in Fig. 13, the gap between the model's training and validation accuracy rate closes

at around 95%. Therefore the model training is stopped after this point to avoid overfitting. Training operation is repeated ten times with a randomly selected batch of the training set each time, and the model with the highest accuracy rate was saved and used in the testing process. The final model with the highest offline testing performance is transferred to run on the edge unit for real-time inference.

Offline testing of the model is performed on randomly collected data sets over different days with independent experiments to investigate the robustness of diagnostic accuracy and reproducibility.

Table 3 summarizes the testing accuracies where the best model out of the training phase is transferred to run multiple times for each test belonging to a single class label. Each individual test is repeated ten times, and the average and standard deviations of accuracies are presented in the corresponding row.

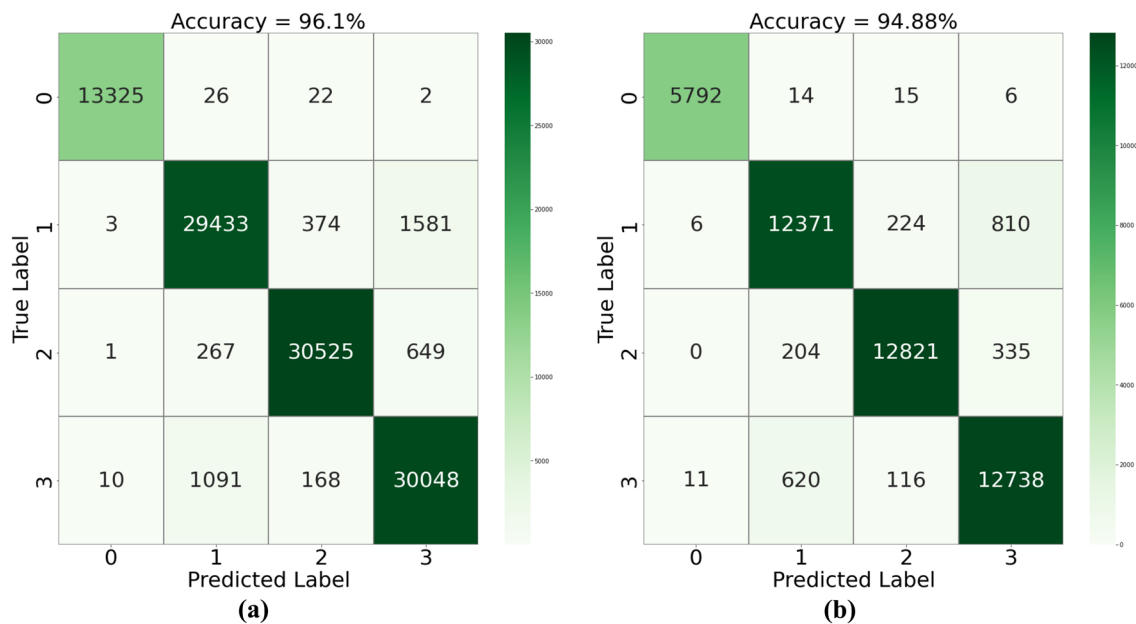


Fig. 12 Confusion matrix **a** training, **b** validation (0: healthy bearing, 1:outer race bearing fault, 2:inner race bearing fault, 3:ball bearing fault)

Table 2 Training and validation performance

	Training				Validation			
	Precision	Recall	f1	Support	Precision	Recall	f1	Support
0	1.00	1.00	1.00	13,375	1.00	0.99	1.00	5827
1	0.96	0.94	0.95	31,391	0.94	0.92	0.93	13,411
2	0.98	0.97	0.98	31,442	0.97	0.96	0.97	13,360
3	0.93	0.96	0.94	31,317	0.92	0.94	0.93	13,485
Accuracy			0.96	107,525			0.95	46,083
Macro avg	0.97	0.97	0.97	107,525	0.96	0.96	0.96	46,083
Weighted avg	0.96	0.96	0.96	107,525	0.95	0.95	0.95	46,083

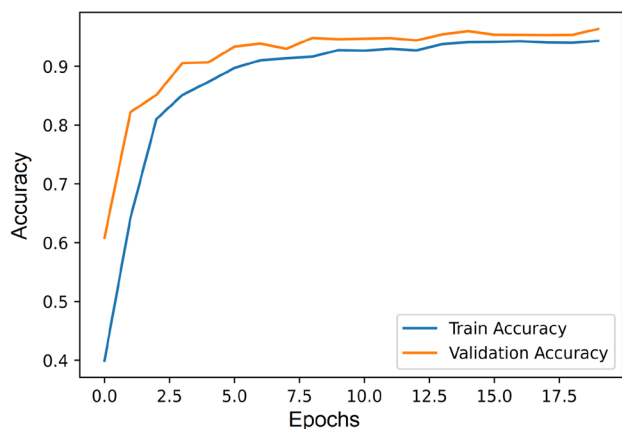


Fig. 13 Training performance of the DL model

Table 3 Test data accuracy and standard deviations (STD)

Test no	Class label	Accuracy % (avg)	f1	STD
1	Healthy (0)	87.21	0.87	1.7394
2	Healthy (0)	90.92	0.91	1.7606
3	Healthy (0)	90.29	0.90	0.2050
4	Healthy (0)	92.20	0.92	5.9043
5	Outer (1)	93.05	0.93	0.3959
6	Outer (1)	95.15	0.95	0.4737
7	Inner (2)	95.34	0.95	0.1555
8	Inner (2)	91.49	0.91	0.8768
9	Ball (3)	83.57	0.83	4.9426
10	Ball (3)	89.69	0.90	0.1979

Figure 14 presents a confusion matrix for test data where the model is applied to predict the four classes simultaneously. A total of 6400 test data belonging to four classes were applied to the model. 1599 of 1625 data belonging to healthy bearing were determined correctly. Three were predicted as outer race fault, ten as inner race fault, and 13 as ball fault. Of the 1548 data belonging to the outer race faulty bearing, 1508 were determined correctly. Four were predicted as healthy, one as inner race fault, and 35 as ball fault. 1504 out of 1584 data of bearing with inner race faulty bearing have been determined correctly. Five were predicted as healthy, 19 as outer race fault, and 56 as ball fault. 1479 of 1643 data belonging to faulty ball bearings were determined correctly. Forty-three of them were found to be healthy, 104 of them as outer race fault, and 17 of them as inner race fault.

The results show that the model can have significantly high true positive rates, indicating a satisfactory model performance. As the last step, the final model is transferred to run on the edge device and observed to show successful behavior in detecting bearing faults.

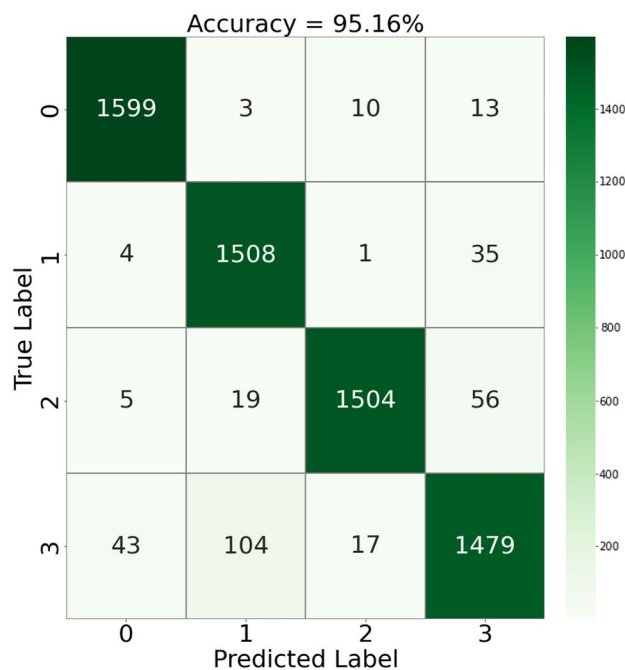


Fig. 14 Test confusion matrix

Discussion

During experimental studies, each test is repeated ten times to ensure reproducibility of the model classification performance. The tests were carried out during motor operations under four different loading conditions. These conditions are 25%, 50%, 75%, and 100%. One-hundred percent loading condition corresponds to where the motor is operated under maximum load. Loading conditions are repeated for each class type, including healthy, outer, inner, and faulty ball bearings. The varying loading conditions of the motor under the same bearing type are expected to cause slight deviations in the vibrational signal characteristics. Therefore, we consider that this might contribute to the varying standard deviations given in Table 3.

The confusion matrix (belonging to four classes) with an accuracy rate of 95.16% is given in Fig. 14. 1479 of 1643 data belonging to faulty ball bearings were determined correctly. One hundred four of them were predicted as outer race fault. This is an expected result. The faulty bearing ball rolling inside the bearing physically contacts both the inner and outer races, simultaneously. This might cause the two classes to become slightly convoluted. Therefore the accuracy rate of the prediction of ball fault is observed to be comparatively lower.

Conclusion

In this study, unlike the majority of the studies in the literature focusing on algorithm development with standard datasets for motor faults, a new induction motor condition monitoring system based on IoT technologies is designed and implemented. An end-to-end data pipeline is experimentally tested on transferring multiple time-series sensor data from the Edge to database storage for long-term data persistence. The monitoring system is demonstrated on a use case to detect and classify induction motors' bearing faults by mainly focusing on processing vibration signals and utilizing DL-based models that can be deployed and executed at the Edge units.

The Edge unit consists of an industrial-grade data acquisition NI CompactRIO hardware and is programmed to process monitoring signals in LabVIEW. A new custom-made testbed design is introduced to generate artificially induced fault scenarios and develop DL-based fault detection and diagnosis models. It is experimentally verified that a CNN-based DL algorithm can be successfully integrated into the presented monitoring system to detect and diagnose various bearing faults.

In future, this monitoring system can easily be scaled to monitor fleets of induction motors in the field. The vibration signals may be recorded for longer periods of time. Utilizing longer-term historical data collection, prognostics studies can be easily implemented to estimate the remaining useful times of each piece of equipment. With the help of the data analytics module's highly customizable features, dashboards can be realized to monitor subscribed equipment motor status over the internet remotely. By creating rule-based triggering mechanisms based on monitored signals and the Edge-AI model's inference results, warning notifications can be delivered to system stakeholders. If a critical level of fault is detected, a halt signal can be sent to the control relay circuitry to cut the malfunctioning motor's power immediately.

Funding This research is supported in part by the Scientific and Technical Research Council of Turkey (TUBITAK) under 2232 International Fellowship for Outstanding Researchers Program with the grant number 118C252.

Declarations

Conflict of Interest The authors declare that no conflict of interest exists in the submission of this article.

References

- Anna S, Ottewill JR, Baranowski J, Orkisz M (2019) A PCA and two-stage bayesian sensor fusion approach for diagnosing electrical and mechanical faults in induction motors. *IEEE Trans Ind Electron* 66:9510–9520. <https://doi.org/10.1109/TIE.2019.2891453>
- Ali MZ, Shabbir MNSK, Liang X et al (2019) Machine learning-based fault diagnosis for single- and multi-faults in induction motors using measured stator currents and vibration signals. *IEEE Trans Ind Appl* 55:2378–2391. <https://doi.org/10.1109/TIA.2019.2895797>
- Ali MZ, Shabbir MNSK, Zaman SMK, Liang X (2020) Single- and multi-fault diagnosis using machine learning for variable frequency drive-fed induction motors. *IEEE Trans Ind Appl* 56:2324–2337. <https://doi.org/10.1109/TIA.2020.2974151>
- Aliev K, Antonelli D, Bruno G, et al (2021) Building system diagnostics through a network of smart local sensors. In: 2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNM). IEEE, Preveza, Greece, pp 1–6
- Carvalho TP, Soares FA, Vita R et al (2019) A systematic literature review of machine learning methods applied to predictive maintenance. *Comput Ind Eng* 137:106024. <https://doi.org/10.1016/j.cie.2019.106024>
- Choudhary A, Jamwal S, Goyal D et al (2020) Condition monitoring of induction motor using internet of things (IoT). In: Kumar H, Jain PK (eds) Recent advances in mechanical engineering. Springer, Singapore, pp 353–365
- Dhiraj N, Seok J (2020) Bearing fault detection and diagnosis using case western reserve university dataset with deep learning approaches: a review. *IEEE Access* 8:93155–93178. <https://doi.org/10.1109/ACCESS.2020.2990528>
- Gang Y (2020) A concentrated time-frequency analysis tool for bearing fault diagnosis. *IEEE Trans Instr Meas* 69:371–381. <https://doi.org/10.1109/TIM.2019.2901514>
- Hoang DT, Kang HJ (2020) A motor current signal-based bearing fault diagnosis using deep learning and information fusion. *IEEE Trans Instr Meas* 69:3325–3333. <https://doi.org/10.1109/TIM.2019.2933119>
- Hosseinpoor Z, Arefi MM, Razavi-Far R et al (2021) Virtual sensors for fault diagnosis: a case of induction motor broken rotor bar. *IEEE Sens J* 21:5044–5051. <https://doi.org/10.1109/JSEN.2020.3033754>
- Kumar Jalan A, Patil S, Mittal G (2020) A review on fault diagnosis of misaligned rotor systems. *Int J Perform Eng* 16:499. <https://doi.org/10.23940/ijpe.20.04.p1.499509>
- LabView (2021) National Instruments
- Lacatus P, Ionel M, Matache G, Barbu V (2021) Strategy for implementation and use of monitoring systems for gear-motors. *Hidraulica Magazine*. <https://hidraulica.fluidas.ro/2021/nr3/86-93.pdf>. Accessed 1 July 2022
- Lei Y, Yang B, Jiang X et al (2020) Applications of machine learning to machine fault diagnosis: a review and roadmap. *Mech Syst Signal Process* 138:106587. <https://doi.org/10.1016/j.ymsp.2019.106587>
- Li G, Deng C, Wu J et al (2020) Rolling bearing fault diagnosis based on wavelet packet transform and convolutional neural network. *Appl Sci* 10:770. <https://doi.org/10.3390/app10030770>
- Lu S, Qian G, He Q et al (2020) In situ motor fault diagnosis using enhanced convolutional neural network in an embedded system. *IEEE Sens J* 20:8287–8296. <https://doi.org/10.1109/JSEN.2019.2911299>
- Pakpahan MDF, Crisgar PV, Wijaya PR, et al (2021) Implementation of Data communication for motor vehicle tracking systems. In: 2021 International Symposium on Electronics and Smart Devices (ISESD). IEEE, Bandung, Indonesia, pp 1–5
- Shao S, Yan R, Lu Y et al (2020) DCNN-based multi-signal induction motor fault diagnosis. *IEEE Trans Instr Meas* 69:2658–2669. <https://doi.org/10.1109/TIM.2019.2925247>

19. Tang H, Lu S, Qian G et al (2021) IoT-based signal enhancement and compression method for efficient motor bearing fault diagnosis. *IEEE Sens J* 21:1820–1828. <https://doi.org/10.1109/JSEN.2020.3017768>
20. Wang D, Li Y, Jia L et al (2021) Novel three-stage feature fusion method of multimodal data for bearing fault diagnosis. *IEEE Trans Instr Meas* 70:1–10. <https://doi.org/10.1109/TIM.2021.3071232>
21. Wang J, Du G, Zhu Z et al (2020) Fault diagnosis of rotating machines based on the EMD manifold. *Mech Syst Signal Process* 135:106443. <https://doi.org/10.1016/j.ymssp.2019.106443>
22. Xu Y, Li Z, Wang S et al (2021) A hybrid deep-learning model for fault diagnosis of rolling bearings. *Measurement* 169:108502. <https://doi.org/10.1016/j.measurement.2020.108502>
23. Yang J, Yin S, Chang Y et al (2021) An efficient method for monitoring degradation and predicting the remaining useful life of mechanical rotating components. *IEEE Trans Instr Meas* 70:1–14. <https://doi.org/10.1109/TIM.2020.3032218>
24. Yuan L, Lian D, Kang X et al (2020) Rolling bearing fault diagnosis based on convolutional neural network and support vector machine. *IEEE Access* 8:137395–137406. <https://doi.org/10.1109/ACCESS.2020.3012053>
25. Zhang S, Zhang S, Wang B, Habetler TG (2019) Deep learning algorithms for bearing fault diagnostics: a review. In: 2019 IEEE 12th International Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives (SDEMPED). IEEE, Toulouse, France, pp 257–263

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Authors and Affiliations

Eyüp Irgat¹  · Eyüp Çınar^{3,4} · Abdurrahman Ünsal² · Ahmet Yazıcı^{3,4}

Eyüp Çınar
eyup.cinar@ogu.edu.tr

Abdurrahman Ünsal
unsal@dpu.edu.tr

Ahmet Yazıcı
ayazici@ogu.edu.tr

¹ Department of Kutahya Technical Sciences Vocational School, Kutahya Dumlupinar University, Evliya Celebi Campus, 43100 Kutahya, Turkey

² Department of Electrical Electronics Engineering, Kutahya Dumlupinar University, Evliya Celebi Campus, 43100 Kutahya, Turkey

³ Department of Computer Engineering, Eskisehir Osmangazi University, Meselik Campus, 26040 Eskisehir, Turkey

⁴ Center of Intelligent Systems Applications and Research (CISAR), Eskisehir Osmangazi University, Eskisehir, Turkey