SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I6P124

Volume 11 Issue 6, 224-233, June 2024

Original Article

Revolutionizing Conveyor Belt Systems: Empowering Predictive Maintenance with IoT, Cloud, and Machine Learning

P.V.S. Anusha[1], P. Swapna[2], D.V. Rama Koti Reddy[3]

[1,2,3]*Department of Instrumentation Engineering, College of Engineering, Andhra University, Andhra Pradesh, India.*

[2]*Corresponding Author : dr.pswapna@andhrauniversity.edu.in*

Received: 11 April 2024 Revised: 14 May 2024 Accepted: 11 June 2024 Published: 29 June 2024

Abstract *- The ever-increasing volume of data necessitates an effective monitoring system to support decision-making processes. In the Industry 4.0 landscape, Artificial Intelligence (AI) is reshaping manufacturing by leveraging Internet of Things (IoT) technologies and machine learning methods. In this paper, a data-driven predictive maintenance system for conveyor belt systems in manufacturing using IoT technologies and machine learning methods is proposed. The system uses real-time data from IoT devices such as accelerometers, temperature, and current sensors deployed on conveyor belts integrated with ESP32 and AWS cloud infrastructure. The study evaluates the efficacy of the developed predictive maintenance system using real-world IoT data from manufacturing environments and machine learning. The top-performing algorithm is the extra trees classifier with the highest accuracy, which shows superior performance across multiple metrics. The results demonstrate the system's success in identifying potential failure indicators, thereby mitigating production downtimes. The paper highlights the significance of the belt conveyor system in various industries and the need for efficient maintenance methods to ensure smooth operation.*

Keywords *- Internet of Things (IoT), Machine Learning, Conveyor belt, Predictive maintenance, Extra trees.*

1. Introduction

Belt conveyor systems are highly effective in transporting a diverse range of materials, including those with varying shapes, sizes, and weights, economically and expeditiously. These systems are noteworthy for their capacity to convey materials directly on the belt surface, making them an indispensable component in industries that handle bulk materials. In manufacturing, transportation, warehouses, and other contexts where the movement of bulk materials is necessary, belt conveyor systems play a crucial role [1, 2].

The conveyor belt system is a vital component in bulk material handling systems, characterized by the presence of rollers and chains that can be customized to meet specific requirements. The selection of conveyor systems depends on factors such as material size, distance, and speed requirements. The ongoing demand for increased tonnages over longer routes has prompted technological advancements in the system's design, analysis, and simulations.

The extensive applications of the belt conveyor system extend to various industries, including mining, cement, power plants, and other production sectors. Different design parameters are employed to cater to specific application contexts, such as coal mines, cement, and the food industry [3]. In the realm of material handling systems, the efficiency of conveyor systems greatly depends on the proper functioning of essential components, including belts, pulleys, and electric motors. Traditional methods of detecting faults in these systems have proven to be unreliable and time-consuming, necessitating extensive maintenance efforts. Conveyor belt mis-tracking, slippage, and seized rollers are common issues that can result in unexpected downtime and maintenance challenges.

To alleviate these challenges, the use of belt sway switches as a preventive measure against mis-tracking and potential malfunctions is recommended. To address slippage concerns, it is necessary to remove material build-up and adjust tension to prevent excessive stretching and audible squeals. Regular maintenance is essential in avoiding the consequences of seized rollers, which can lead to mis-tracking and other operational issues [4].

The Conveyor belt system plays a vital role in numerous industries, yet traditional methods for identifying faults in these systems are often untrustworthy and lengthy, requiring extensive maintenance efforts. Thus, an efficient method for predicting faults in conveyor belt systems is sorely needed. In steel plants, where each component plays a vital role, the implementation of a dedicated condition monitoring system for conveyors emerges as a strategic solution with the potential to enhance their reliability and overall operational efficiency significantly [1]. The research highlights a gap in the efficiency of existing fault detection methods within conveyor systems. This gap underscores the need for advancements in fault detection techniques. The problem statement emphasizes the crucial importance of enhancing the reliability of conveyor systems to ensure smooth operations across various industries. Addressing this imperative is essential for optimizing productivity and minimizing downtime in industrial processes reliant on conveyor systems.

2. Literature Review

The Internet of Things (IoT) has significantly impacted industries by enabling the connection of a multitude of devices to cloud services, thereby offering cost-effective solutions. By utilizing the internet, IoT facilitates the interconnection of a diverse range of electrical appliances and objects, thereby forming a network of hardware, software, and data storage systems. The Internet of Things can be applied in the domains of manufacturing, industrial automation, and machine condition monitoring [5].

The application of Industry 4.0 and IoT in predictive maintenance and reliability enhancement has been a focal point, particularly in manufacturing, yet research on applying these concepts to baggage handling systems remains limited [6]. Studies have delved into transforming airport baggage systems by emphasizing continuous maintenance improvement using condition monitoring techniques, specifically at Heathrow's Destination-Coded Vehicle (DCV) baggage cart system [6].

Companies like JFE Steel have developed innovative belt conveyor monitoring systems utilizing ICT at their raw material yard, employing wireless networks for large-scale data collection and an image judgment system to detect belt shape defects. Early detection of abnormalities is crucial, particularly in integrated steel plants, due to varying equipment levels, necessitating different maintenance needs [7].

Moreover, the integration of IoT technology in belt conveyor systems has paved the way for various advancements, such as a smart belt conveyor speed control system that employs embedded web and microcomputer technologies for monitoring and adjusting speeds between conveyor stages, reducing energy loss [8]. Similarly, innovative online diagnostic technologies based on motor current signature analysis have demonstrated effectiveness in detecting belt mistracking issues, offering promise for diverse industries like airports, mining, manufacturing, and transportation [9]. The integration of Machine Learning (ML) models has significantly impacted fault detection and monitoring systems in conveyor belts. Studies have showcased the utilization of Radio Frequency Identification (RFID) sensing technology in crack detection in conveyor belts, particularly in mining operations. Structural Health Monitoring (SHM) models using machine learning and IoT connectivity have shown high accuracy in detecting crack attributes, ensuring safe and efficient operations [10]. IoT technology has also been instrumental in maintaining optimal yarn quality in spinning mills, integrating various sensors to oversee moisture and temperature levels, ultimately reducing yarn breakage and improving quality [11].

Additionally, IoT-based fault diagnosis systems have been developed for conveyor belts in coal production, offering timely warnings to prevent accidents and revolutionize fault diagnosis in the coal industry [12]. Machine learning has revolutionized maintenance strategies by leveraging predictive analytics to avoid production disruptions due to equipment failures [13]. Computer vision, employed to monitor conveyor belt alignment, has shown promising results, offering safer and more efficient conveyor operations [14]. Furthermore, fault detection in belt conveyor idlers using sound signals and machine learning models has demonstrated high accuracy in diagnosing issues [15].

The adoption of a novel maintenance decision-making framework for belt conveyor idlers has shown improved predictability and accuracy in diagnosing idler roll failures, calling for further research to refine inspection interval models [16]. Machine learning methods have been employed to create classification models for assessing damage in rubber textile conveyor belts, indicating the potential efficiency of using repaired or renovated belts [17].

Moreover, the application of Distributed Optical Fibre Sensors (DOFS) for monitoring mining conveyors has exhibited promising capabilities in detecting and classifying damage under various conveyor speeds, contributing to economic loss prevention and enhanced personnel safety [18]. The integration of IoT and machine learning in fault detection systems for conveyor belts across various industries has shown immense potential in improving operational efficiency and ensuring safety [10, 12, 18].

In summary, the combination of IoT, machine learning, and Industry 4.0 has had a significant impact on the improvement of conveyor belt maintenance and fault detection in various industries, resulting in the emergence of predictive maintenance and improved operational performance. This paper proposes the development of a predictive maintenance system for conveyor belts that incorporates IoT, Cloud, and machine learning technologies to monitor crucial parameters such as vibration, temperature, and current measurements. This system aims to proactively identify potential faults, enabling timely maintenance interventions and improving the dependability and efficiency of conveyor belt systems in industrial environments.

3. Materials and Methodology

3.1. Predictive Maintenance System Architecture

A method for continuously monitoring the health of a conveyor belt system has been developed, which makes it easy to maintain the system by monitoring its temperature, vibration, and current. The overall system architecture of the conveyor belt predictive maintenance system is shown in Figure 1. The proposed system comprises three main components:

- A sensor node design that includes an ESP32 development board, vibration sensor, current sensor, and temperature sensor for gathering sensor data;
- Amazon Web Services (AWS) for collecting, storing, and analysing the data;
- And machine learning algorithms for developing a conveyor belt fault prediction model.

The data flow process begins with the ESP32 development board, which records temperature, vibration, and current readings from the sensors connected to it. The ESP32 then sends this data to AWS IoT Core via Wi-Fi, using the MQTT messaging protocol. The data is stored in AWS DynamoDB. This data is then used to create a dataset, which is used to build a machine learning model for predicting conveyor belt faults.

3.2. Sensor Node Design

The construction of the sensor node entails the integration of three components: the LIS2DW12 accelerometer sensor, the DFRobot Gravity I2C Non-contact IR temperature sensor for Arduino (MLX90614-DCC), and the WCS1700 hall current sensor. These sensors were integrated with an ESP32 development board, and a schematic diagram depicting this process is provided in Figure 2. The acquisition of sensor data from the ESP32 was facilitated by programming the firmware using the Arduino Integrated Development Environment.

3.2.1. ESP32 Development Board

The ESP32 development board serves as the central component of the system, functioning as a high-performance microcontroller that is widely utilized in IoT, smart home automation, and embedded systems. The ESP32 is distinguished by its dual-core architecture, expanded GPIO pins, and enhanced cryptographic capabilities, making it a suitable replacement for the ESP8266.

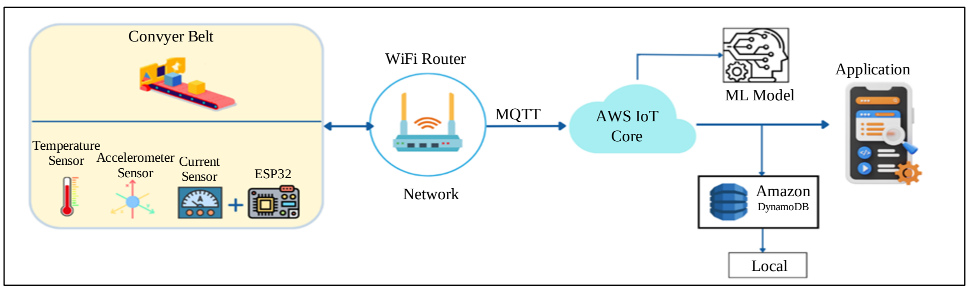


Fig. 1 Architecture of conveyor belt predictive maintenance system

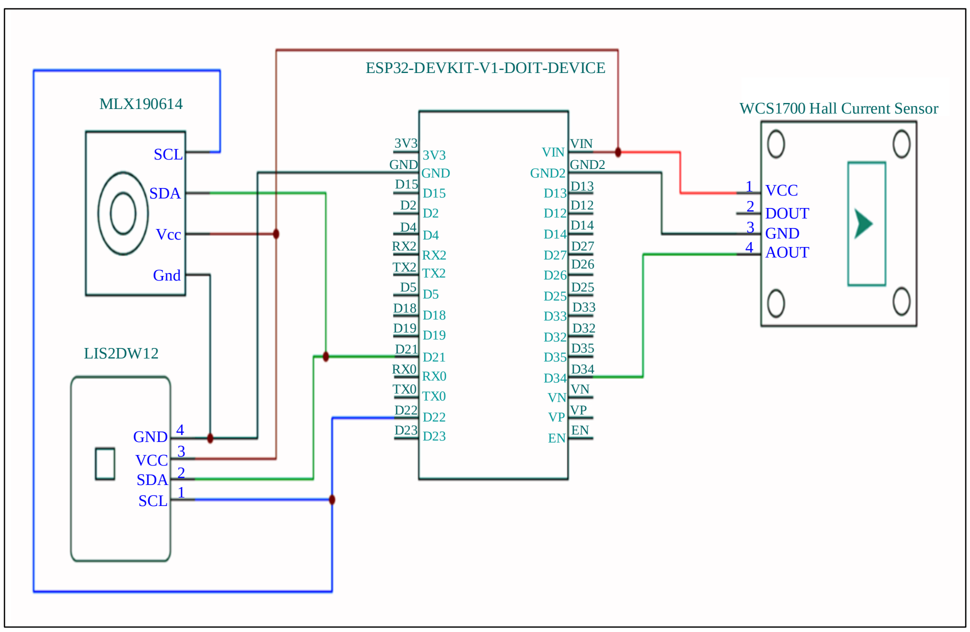


Fig. 2 IoT device schematic representation

The ESP32 offers a multitude of built-in sensors and supports FreeRTOS, enabling seamless integration with the Arduino platform and facilitating rapid prototyping and task-oriented processing. This study focuses on the potential of the ESP32 to present real-time data on various screens rather than its conventional web server applications. The versatility and extensive range of applications of the ESP32 make it a promising area for innovative exploration across multiple industries and research domains [19, 20].

3.2.2. LIS2DW12 Accelerometer Sensor

The LIS2DW12 accelerometer sensor incorporates MEMS technology and capacitive sensing to detect acceleration in three axes (X, Y, Z). It measures changes in capacitance due to movement and converts them into a digital signal, which it communicates to a host system via the I2C or SPI protocols. This high-precision acceleration data is suitable for applications such as motion detection, orientation sensing, and impact analysis. Its configurable settings make it adaptable to various use cases. To interface the LIS2DW12 accelerometer sensor with the ESP32 using the I2C protocol, it is necessary to link the SDA and SCL pins of the LIS2DW12 to the corresponding pins on the ESP32. The LIS2DW12 has a default I2C address of 0x19. By utilizing the ESP32's I2C library and functions, the microcontroller can initialize communication by specifying the I2C address of the LIS2DW12 (0x19 in this instance) and then request acceleration data using specific commands. The ESP32 can then receive and interpret the data transmitted by the LIS2DW12 accelerometer sensor. This method demonstrates the steps involved in establishing a connection between the ESP32 and the LIS2DW12 accelerometer via the I2C protocol, allowing for the retrieval and utilization of acceleration data for various applications [21].

3.2.3. Non-Contact IR Temperature Sensor

The DFRobot gravity I2C non-contact IR temperature sensor, which is equipped with the MLX90614-DCC, is a sophisticated and adaptable device designed for precise, non-contact temperature measurement. This sensor employs infrared technology to accurately measure temperatures without any physical contact with the target object. The I2C interface also enables seamless compatibility with Arduino and other microcontrollers, allowing for easy integration into various applications, including industrial temperature monitoring, thermal imaging, and non-invasive temperature measurement in medical and consumer electronic devices. This sensor serves as a reliable and efficient solution for temperature sensing and caters to a wide range of research and development needs. The ESP32 microcontroller was successfully interfaced with the MLX90614-DCC IR temperature sensor using the I2C protocol. By connecting the SDA and SCL pins of the MLX90614 to the corresponding pins on the ESP32 and recognizing the sensor's default I2C address of 0x5A, the ESP32 was able to communicate effectively to request and retrieve non-contact temperature measurements. This integration highlights the potential for diverse applications using non-contact temperature sensing and demonstrates a promising avenue for further research in sensor integration and data acquisition methodologies [22, 23].

3.2.4. WCS1700 Hall Current Sensor

The WCS1700 Hall Current Sensor with over current protection is a groundbreaking innovation in the realm of current sensing technology. This sensor, based on the Hall Effect principle, offers precise and non-invasive measurement of current flow within electrical systems. The integrated over-current protection feature ensures the safety and longevity of the monitored systems by promptly detecting and responding to excessive current levels.

The WCS1700 sensor is a robust solution for accurate current measurement in various applications, including power supplies, motor control, and energy management systems. Its ability to accurately measure current, along with built-in protective mechanisms, makes it an indispensable tool for research, addressing the need for reliable, high-performance current sensing in diverse industrial and scientific settings. By connecting the ESP32 microcontroller with the WCS1700 hall current sensor by linking their respective signal and power pins, the ESP32 is able to interpret the sensor's output.

The ESP32's Analog-to-Digital Converter (ADC) functionality is employed to efficiently read the WCS1700's output, converting it into measurable current values. This integration showcases the WCS1700's capacity for precise current sensing and its potential applications in energy monitoring and power management. Upon interfacing the sensors and uploading the program into the esp32 using the Arduino IDE, the sensor data is obtained in the serial monitor, as illustrated in the accompanying Figure 3.

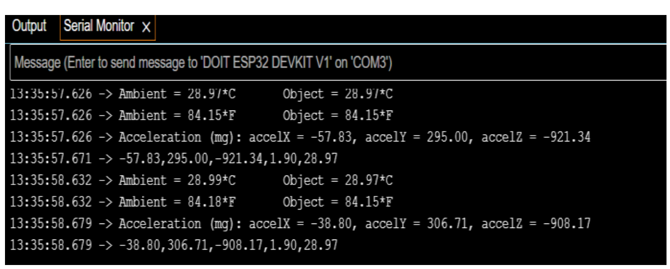


Fig. 3 Visualization of data in Arduino serial monitor

3.3. AWS Cloud Platform

Once the ESP32 has acquired sensor data, it must connect to the internet using Wi-Fi to enable the transmission of this data to the cloud. Upon establishing an internet connection, the ESP32 transmits the sensor data to Amazon Web Services (AWS) IoT Core using the MQTT protocol. AWS IoT core provides five key services tailored to IoT devices, ensuring seamless connectivity, cloud-based management, OTA updates, and robust security measures. These core services encompass rules, topics, shadow service, AWS IoT device defender, and AWS IoT device management.

The rules functionality within AWS IoT core regulates IoT device behavior by enabling integration with various AWS services. These rules enable various functionalities, such as data filtering, categorization, real-time processing, and triggering alerts based on specified thresholds or exceptional events. The MQTT protocol is used in the AWS IoT core system, which offers a flexible and efficient process where rules are analysed, actions are performed, and messages are routed based on topic streams.

Regarding security measures, AWS IoT device defender plays a crucial role by assigning digital identities to IoT devices, guaranteeing authentication, authorization, and encryption. It conducts ongoing monitoring of IoT device configurations, promptly alerting users and services to any deviations from the anticipated behavior. Additionally, AWS IoT device management enables remote management of IoT devices, effectively handling issues related to device behavior anomalies by managing and updating firmware and software.

The AWS IoT Core framework offers a robust foundation for the secure, scalable, and efficient management of IoT devices, regardless of the specific application domain. AWS IoT core acts as an intermediary to efficiently store the sensor data in a DynamoDB table [24]. DynamoDB, a cloud-based NoSQL database service managed by Amazon Web Services (AWS), securely maintains the data within the AWS infrastructure, ensuring both accessibility and scalability.

This AWS infrastructure provides a dependable and sturdy platform for managing and processing incoming sensor data. Thereafter, the collected data from DynamoDB can be accessed for analysis, visualization, and further processing. Researchers and developers can extract this stored information for various applications, such as data analytics, visualization tools, statistical analysis, and the development of machine learning models. The integration of AWS IoT Core and DynamoDB presents an efficient method for securely storing and managing sensor data in a scalable cloud environment. This methodology not only focuses on organizing and storing data but also emphasizes its seamless retrieval and utilization for a wide range of analytical, visualization, and machine-learning purposes.

3.4. Experimental Setup for Conveyor System Evaluation

The experimental setup was carefully designed with precise specifications in order to ensure optimal performance, as shown in Figure 4. The integration of the ESP32 development board, micro USB cable, DC motor with a rectangular gearbox, power supply, bearings, pulley, conveyor stand, conveyor belt, and conveyor test bed was carried out with great care. The motor's power characteristics were specifically chosen to match the conveyor's load requirements, ensuring efficient operation. The bearings, selected for their uniform size, contribute to the system's reliability and longevity. The high-efficiency belt mechanism facilitates smooth motion transfer and enhances overall effectiveness. Our project also features advanced sensor integration, including a triple-axis accelerometer and temperature sensors near the motor and bearing arrangement.

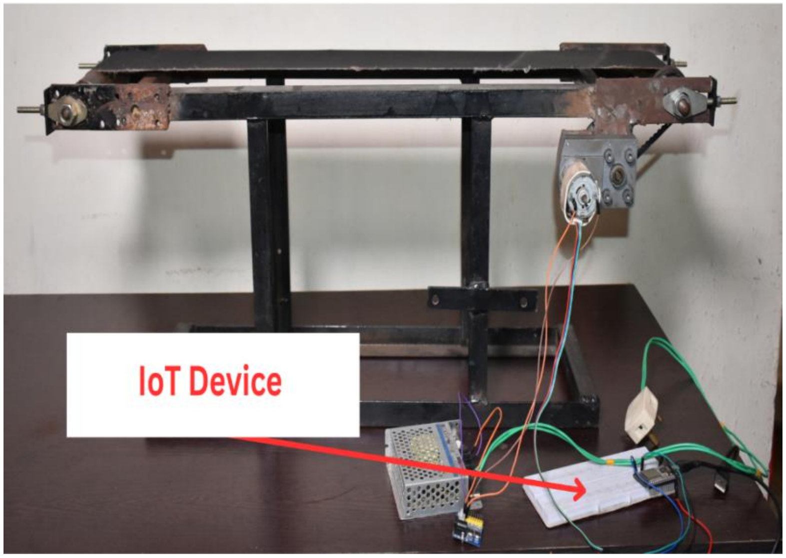


Fig. 4 Experimental setup: conveyor belt system and IoT device testbed

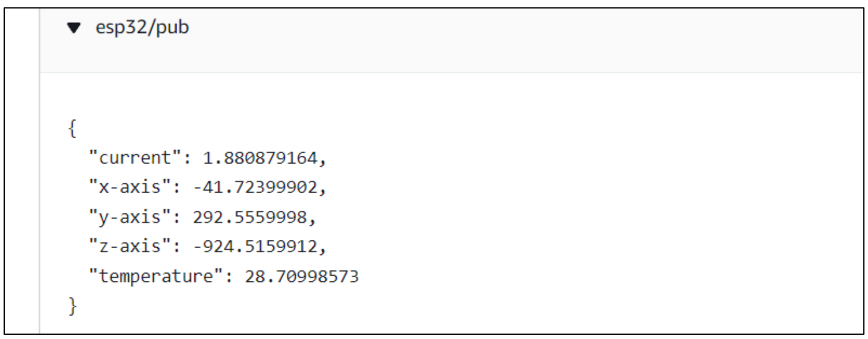
The conveyor system incorporates precise data from vibration, WCS1700 hall current sensor, and temperature sensors, which are processed using Python and essential machine learning algorithms both in real-time and offline. The assembly comprises essential components such as a roller shaft, bolt, bearing chock, and adjusting bracket.

The adjusting bracket securely fastens to the conveyor frame, effectively enclosing the roller shaft within the bearing chock. This arrangement allows for precise adjustments to the roller shaft's position via a bolt connected to the bearing chock through a nut. Such adjustments are critical for ensuring the accurate and controlled functioning of the system.

Our IoT system is integrated with the test bed, which encompasses the physical configuration, including the conveyor system and its constituent components, while the IoT system comprises interconnected devices and software for data collection and analysis. The Internet of Things (IoT) system collects information from the test bed via sensors and transmits it to cloud services, including Amazon Web Services (AWS).

As illustrated in Figure 5, the sensor data from the IoT device is visualized using the MQTT Test client in AWS IoT Core. Developed a real-time node-red dashboard that utilizes an MQTT subscribe node and node-red dashboard nodes. The data is obtained from AWS IoT Core by subscribing to it, which is then in JSON format.

Using function nodes in node-red, the data was separated into temperature current vibration along the X-axis, Y-axis, and Z-axis. The data is then visualized using gauges and chart widgets, as shown in Figure 6.



esp32/pub
{
"current": 1.880879164,
"x-axis": -41.72399902,
"y-axis": 292.5559998,
"z-axis": -924.5159912,
"temperature": 28.70998573
}

Fig. 5 The sensor data in AWS IoT cloud via MQTT test client

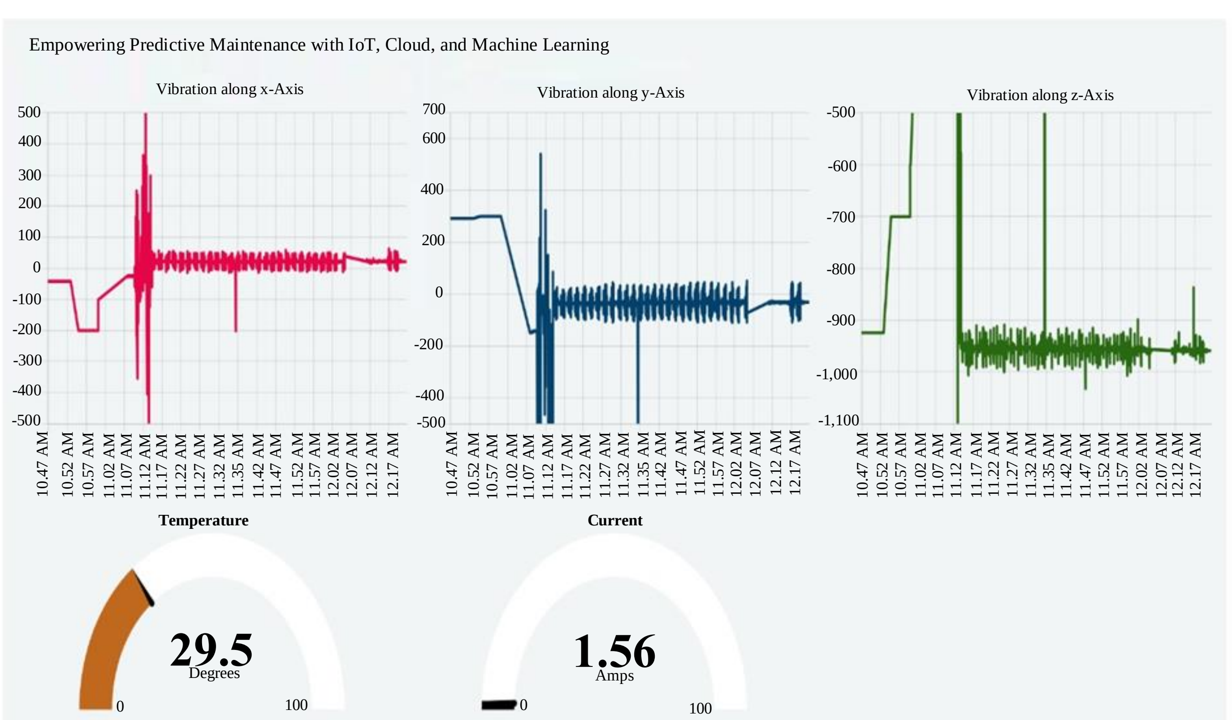


Fig. 6 Visual representation of sensor data in node-red dashboard

The data is subsequently processed stored, and can be utilized for analysis and predictive maintenance. The integration enables real-time data acquisition, remote monitoring, and the fusion of physical and digital data for comprehensive analysis and enhancements in the test bed's functionality. The dataset for this research was obtained from AWS IoT Core and was subsequently loaded into a panda data frame for analysis.

The data was collected with and without fault, which was created using a bearing fault. Preliminary preprocessing steps were performed, which included dividing the data into two subsets, labelled as data1 and data2, and categorized as labels 0 and 1, to form a binary classification dataset. Next, the features and target data were extracted for analysis. A train-test split was performed to create training and testing sets, designated as xtrain, xtest, ytrain, and ytest. An array of diverse machine learning models was employed to address the classification task in this study.

3.5. Machine Learning

The construction of the predictive maintenance system involved the utilization of several machine learning models, including the Decision Tree Classifier (DTC), K-Nearest Neighbors (KNN), Logistic Regression (LRG), Support Vector Machines (SVM) and extra-trees classifier. Decision Tree Classifier (DTC) is a type of tree-like structure in which each internal node signifies a "decision" based on a feature, leading to various branches that represent potential outcomes.

On the other hand, K-Nearest Neighbors (KNN) is a classification algorithm that does not rely on predefined parameters and labels data points according to the majority label of the closest neighboring points in the feature space. Logistic Regression (LRG) is a linear classification algorithm that predicts the probability of a binary outcome based on one or more independent variables. Support Vector Machines (SVM) is a classification method that finds the hyperplane that best separates different classes while maximizing the margin between them in a high-dimensional space. Finally, the extra-trees Classifier is an ensemble learning technique that improves accuracy and controls over-fitting by averaging the predictions of multiple randomized decision trees fitted on different subsets of the dataset.

4. Results and Discussions

Multiple machine learning models were employed for the accurate classification of conveyor belt fault detection, including decision tree classifier, K-nearest neighbors, logistic regression, support vector machines, and extra-trees classifier. These models were trained using the Scikit-learn framework. The study aimed to assess the performance metrics of these classifiers across different datasets, including accuracy, balanced accuracy, precision, recall, F1 score, and ROC AUC score. Confusion matrices were generated to assess each model's classification performance. Table 1 provides an accurate comparison of the various classification models' performance, including their precision, recall, F1 score, and ROC AUC.

Bar plots were utilized to visually compare the performance metrics of the trained classifiers, providing a clear understanding of how each model performed in the context of conveyor belt classification. In this study, a comprehensive evaluation was conducted to assess the performance of various machine learning models. The results, as depicted in Figure 7, indicate a remarkable trend in the accuracy of these models. The decision tree classifier and the KNN models demonstrated noteworthy performance in our classification task, with accuracy scores of 98.765% and 96.26%, respectively. The extra-trees classifier, however, achieved the highest accuracy rate of 100%, showcasing its exceptional capabilities. Meanwhile, the SVM model exhibited an accuracy of 95.06%. These results act as an essential benchmark, offering a detailed assessment of each model's performance. The findings emphasize the superior accuracy of several models while also highlighting the varying levels of performance across the models assessed. The results indicate that the Decision Tree Classifier (DTC) and Support Vector Machines (SVM) achieved the highest precision scores of 0.97368 and 0.94595, respectively.

Table 1. The summarized model outcomes

Classification Model	Accuracy Score	Balanced Accuracy	Precision	Recall	F1 Score	ROC_AUC
Decision Tree Classifier (DTC)	0.98765	0.9886	0.97368	1.0	0.98667	0.9886
K-Nearest Neighbour (KNN)	0.96296	0.9616	0.97222	0.94595	0.9589	0.9616
Logistic Regression (LRG)	0.81481	0.8123	0.80556	0.7837	0.79452	0.8123
Support Vector Machines (SVM)	0.95062	0.9502	0.94595	0.94595	0.94595	0.9502
Extra Trees Classifier (EXT)	1.0	1.0	1.0	1.0	1.0	1.0
Naïve Bayes Classifier (NVB)	0.92593	0.9275	0.89744	0.94595	0.92105	0.9275

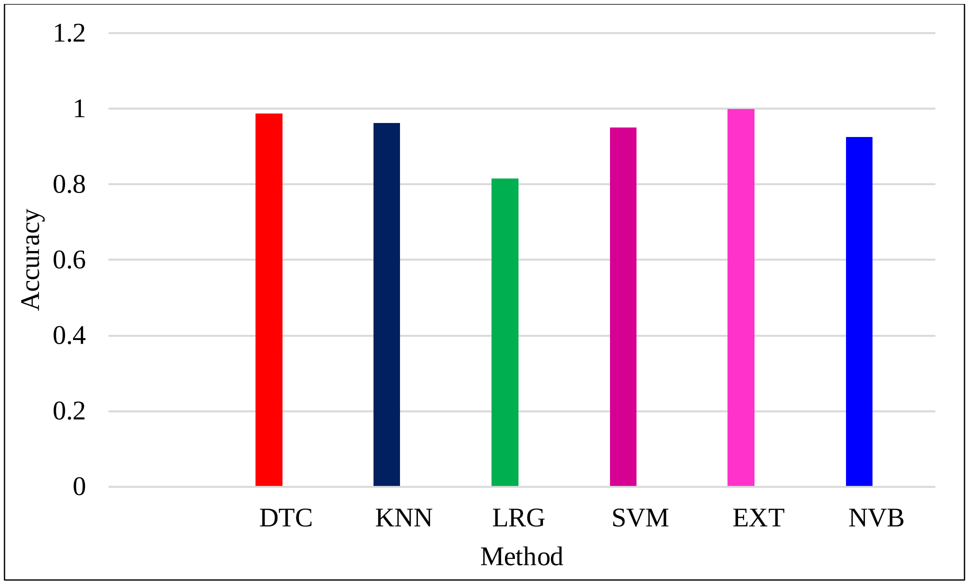


Fig. 7 Accuracies of different machine learning models

Conversely, the Logistic Regression (LRG) model obtained the lowest precision score of 0.80556. The Extra Trees (EXT) model demonstrated perfect precision, while the Naive Bayes (NVB) model attained a moderate precision score of 0.89744. Furthermore, the table showcases the recall scores of different models, with the DTC and EXT models achieving perfect recall scores of 1.0. The K-Nearest Neighbor (KNN) and SVM models exhibited a recall score of 0.94595, while the LRG model had a slightly lower recall score of 0.7837. The F1 scores of the models are also displayed, with the DTC, KNN, SVM, and NVB models being the top performers.

The DTC achieved an impressive F1 score of 0.98667, the KNN model demonstrated a commendable score of 0.9589, the SVM model displayed its ability with an F1 score of 0.94595, and the NVB classifier attained an impressive F1 score of 0.92105. The scores offer a comprehensive overview of the models' overall performance, highlighting their ability to balance precision and recall effectively. Table 1 provides an in-depth analysis of the ROC AUC scores of various machine learning models, offering significant insights into their capacity to distinguish between positive and negative classes.

Each of the classifiers, including the decision tree classifier, K-nearest neighbor model, logistic regression model, support vector machines model, extra trees classifier, and naive bayes classifier, was subjected to rigorous evaluation. The decision tree classifier achieved the highest ROC AUC score of 0.9886, followed by the K-nearest neighbor model with a score of 0.9616 and the logistic regression model with a score of 0.8123. The support vector machines model and extra trees classifier both attained scores of 0.9502 and 1.0, respectively, reflecting exceptional performance in class discrimination. The naive bayes classifier achieved a score of 0.9275, suggesting reasonable effectiveness in class discrimination. The examination of the performance metrics for the models provides valuable insights into their effectiveness in classifying faults in conveyor belts. The Extra Trees (EXT) model stands out with an impressive precision score, which suggests its ability to classify positive instances without any false positives accurately. This precision is crucial because it helps to minimize the risk of false alarms and unnecessary maintenance interventions, thus ensuring the reliability of the fault detection system. Furthermore, the EXT model achieves a perfect recall score, indicating that it can successfully identify all positive instances, which further affirms its robustness in fault detection.

On the other hand, the Naive Bayes (NVB) model demonstrates moderate precision but still achieves a respectable F1 score, which suggests a balance between precision and recall. Although not as precise as the EXT model, the NVB classifier proves to be effective in detecting positive instances while minimizing false alarms. Additionally, the analysis of the ROC AUC scores reveals the discriminative ability of each classifier in distinguishing between positive and negative classes.

The EXT classifier achieves a perfect score of 1.0, indicating excellent discrimination capability, followed closely by the Decision Tree Classifier (DTC), with a score of 0.9886. These high ROC AUC scores highlight the models' ability to effectively separate fault and non-fault instances, thereby enhancing the overall reliability of the fault detection system. Overall, the comprehensive evaluation of precision, recall, F1 score, and ROC AUC score showcases the strengths of each classifier in balancing accuracy and robustness in fault detection. The EXT model emerges as a top performer, demonstrating exceptional precision, recall, and discrimination capability, making it a promising choice for practical applications in conveyor belt fault detection systems. The Extra Trees Classifier (EXT) demonstrated superior performance in the research due to its ensemble learning approach, which effectively mitigates overfitting and handles noisy data.

By constructing multiple decision trees from random subsets of the training data and selecting the best split at each node, EXT achieves robust generalization to unseen data. Its ability to randomly select thresholds for each feature enhances resilience to noise and outliers, ensuring accurate capture of complex relationships within the dataset while maintaining computational efficiency. These attributes establish EXT as a top performer in classification tasks, making it an ideal choice for predictive maintenance systems, particularly evident in its outstanding performance across multiple evaluation metrics compared to other machine learning models. Therefore, the Extra Trees Classifier emerges as the top performer in the context of the conveyor belt classification project, demonstrating its effectiveness and versatility in real-world applications.

5. Conclusion

The research highlights the significant influence of IoT and AI, particularly in predictive maintenance systems, on the manufacturing industry in the Industry 4.0 era. The predictive maintenance system for conveyor belt systems, utilizing IoT devices and machine learning techniques, has proven to be a valuable resource in proactively identifying potential failures. The study's results show that the system is effective in a real-world manufacturing setting, where it successfully detects warning signs of impending failures, allowing for timely intervention to prevent production disruption.

Table 2 presents a comparative analysis of the proposed methodology and existing approaches is shown in Appendix. The proposed method shows good accuracy compared to other existing methods. The integration of the proposed method into the manufacturing process further enhances the system's reliability and effectiveness. The comprehensive analysis, encompassing metrics such as accuracy, precision, recall, F1 score, and ROC AUC score, provides a nuanced understanding of the models' performance.

The consistent superiority of the extra trees classifier across diverse evaluation criteria solidifies their status as robust and versatile solutions for conveyor belt classification in predictive maintenance systems. In the era of Industry 4.0, with ongoing advancements in manufacturing, this study adds valuable insights to the practical integration of AI-driven solutions. The success of the predictive maintenance system suggests not only improved operational efficiency but also a potential paradigm shift in how manufacturing enterprises approach equipment monitoring and maintenance. The insights gained from this research pave the way for broader applications of similar AI-driven systems, fostering a more resilient and adaptive manufacturing ecosystem.

References

[1] S. Todkar, M. Ramgir, and J.R. Tathwade, "Design of Belt Conveyor System," *International Journal of Science, Engineering and Technology Research*, vol. 7, no. 7, pp. 458-462, 2018. [Google Scholar]

[2] M.A. Alspaugh, "Latest Developments in Belt Conveyor Technology," *MINExpo*, pp. 1-11, 2004. [Google Scholar] [Publisher Link]

[3] Gabriel Fedorko, "Implementation of Industry 4.0 in the Belt Conveyor Transport," *MATEC Web of Conferences*, vol. 263, pp. 1-6, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[4] Gursahib Singh, Manjot Singh, and Arun Kumar, "Application of Conveyor Belt: A Review Paper," *Pramana Research Journal*, vol. 8, no. 8, pp. 452-457, 2018. [Google Scholar] [Publisher Link]

[5] Ranjana Sikarwar, Pradeep Yadav, and Aditya Dubey, "A Survey on IoT Enabled Cloud Platforms," *2020 IEEE 9th International Conference on Communication Systems and Network Technologies*, Gwalior, India, pp. 120-124, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6] Frank Koenig, Pauline Anne Found, and Maneesh Kumar, "Innovative Airport 4.0 Condition-Based Maintenance System for Baggage Handling DCV Systems," *International Journal of Productivity and Performance Management*, vol. 68, no. 3, pp. 561-577, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[7] Ishigaki Yusuke, and Akechi Yoshihiro, "Belt Conveyor Monitoring System for Effective Maintenance Utilizing ICT," *JFE Technical Report*, no. 39-43, 2020. [Google Scholar] [Publisher Link]

[8] Zhiqiang Li et al., "Design of Belt Conveyor Speed Control System of 'Internet+,'" *IOP Conference Series: Materials Science and Engineering*, vol. 563, pp. 1-7, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[9] Len Gelman et al., "Innovative Conveyor Belt Monitoring via Current Signals," *Electronics*, vol. 12, no. 8, pp. 1-9, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[10] Fatema Tuz Zohra et al., "Health Monitoring of Conveyor Belt Using UHF RFID and Multi-Class Neural Networks," *Electronics*, vol. 11, no. 22, pp. 1-24, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[11] T. Kalavathi Devi et al., "IoT Based Moisture Measurement and Conveyor Belt Monitoring in Yarn Mill," *Journal of Physics: Conference Series*, vol. 2325, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[12] Meng Wang et al., "Research on Fault Diagnosis System for Belt Conveyor Based on Internet of Things and the LightGBM Model," *PLoS One*, vol. 18, no. 3, pp. 1-17, 2023. [CrossRef] [Google Scholar] [Publisher Link]
[13] Rafif Nova Riantama et al., "Examining Equipment Condition Monitoring for Predictive Maintenance, A Case of Typical Process Industry," *5th North American International Conference on Industrial Engineering and Operations Management*, pp. 3471-3480, 2020. [Google Scholar] [Publisher Link]
[14] Mengchao Zhang et al., "A Computer Vision Based Conveyor Deviation Detection System," *Applied Sciences*, vol. 10, no. 7, pp. 1-10, 2020. [CrossRef] [Google Scholar] [Publisher Link]
[15] Xiangwei Liu et al., "Acoustic Signal Based Fault Detection on Belt Conveyor Idlers Using Machine Learning," *Advanced Powder Technology*, vol. 31, no. 7, pp. 2689-2698, 2020. [CrossRef] [Google Scholar] [Publisher Link]
[16] Xiangwei Liu et al., "Integrated Decision Making for Predictive Maintenance of Belt Conveyor Systems," *Reliability Engineering & System Safety*, vol. 188, pp. 347-351, 2019. [CrossRef] [Google Scholar] [Publisher Link]
[17] Miriam Andrejiova, Anna Grincova, and Daniela Marasova, "Identification with Machine Learning Techniques of a Classification Model for the Degree of Damage to Rubber-Textile Conveyor Belts with the Aim to Achieve Sustainability," *Engineering Failure Analysis*, vol. 127, 2021. [CrossRef] [Google Scholar] [Publisher Link]
[18] Hendrick Wijaya et al., "Distributed Optical Fibre Sensor for Condition Monitoring of Mining Conveyor Using Wavelet Transform and Artificial Neural Network," *Structural Control and Health Monitoring*, vol. 28, no. 11, 2021. [CrossRef] [Google Scholar] [Publisher Link]
[19] Alexander Maier, Andrew Sharp, and Yuriy Vagapov, "Comparative Analysis and Practical Implementation of the ESP32 Microcontroller Module for the Internet of Things," *2017 Internet Technologies and Applications (ITA)*, Wrexham, UK, pp. 143-148, 2017. [CrossRef] [Google Scholar] [Publisher Link]
[20] Marek Babiuch, Petr Foltýnek, and Pavel Smutný, "Using the ESP32 Microcontroller for Data Processing," *2019 20th International Carpathian Control Conference (ICCC)*, Krakow-Wieliczka, Poland, pp. 1-6, 2019. [CrossRef] [Google Scholar] [Publisher Link]
[21] Martin Pieš, Radovan Hájovský, and Jan Velička, "Wireless Measuring System for Monitoring the Condition of Devices Designed to Protect Line Structures," *Sensors*, vol. 20, no. 9, pp. 1-25, 2020. [CrossRef] [Google Scholar] [Publisher Link]
[22] Agus Sudianto et al., "Smart Temperature Measurement System for Milling Process Application Based on MLX90614 Infrared Thermometer Sensor with Arduino," *Journal of Advanced Research in Applied Mechanics*, vol. 72, no. 1, pp. 10-24, 2020. [CrossRef] [Google Scholar] [Publisher Link]
[23] Gang Jin et al., "Design of Non-Contact Infra-Red Thermometer Based on the Sensor of MLX90614," *The Open Automation and Control Systems Journal*, vol. 7, pp. 8-20, 2015. [Google Scholar]
[24] Kristina Dineva, and Tatiana Atanasova, "Design of Scalable IoT Architecture Based on AWS for Smart Livestock," *Animals*, vol. 11, no. 9, pp. 1-30, 2021. [CrossRef] [Google Scholar] [Publisher Link]

Appendix

Table 2. Comparative analysis of the proposed methodology and existing approaches

Paper	Application	Sensors	ML Algorithms	Accuracy
[10]	ML and IoT-Based Model for Monitoring the Structural Health of Conveyor Belts	RFID based Crack Sensor	Artificial Neural Network (ANN)	95.50%
[12]	An IoT and Machine Learning-Based System for Diagnosing Faults in Belt Conveyors	Speed, Current and Temperature Sensors	Light Gradient Boosting Machine (LGBM) model	97.00%
[15]	Utilizing ML for Fault Detection in Belt Conveyor Idlers Based on Acoustic Signals	Microphone	Gradient Boost Decision Tree	94.53%
Proposed Method	Predictive Maintenance of conveyor belt system using IoT and ML	Temperature, Vibration and Current	Extra Trees Classifier	100%