TELEMETRY DRIVEN PREDICTIVE MAINTENANCE SYSTEM FOR UNMANNED AERIAL VEHICLES USING MACHINE LEARNING

B. Tech. in Aerospace Engineering

Project Work in

Department of Aerospace and Automotive Engineering

Project Team

Sl. No.	Reg. No.	Student Name
1	22ETAS012001	ADITYA ROY

Supervisors : 1. Mr Adarsh Krishnamurthy

January – 2026

FACULTY OF ENGINEERING AND TECHNOLOGY
M. S. RAMAIAH UNIVERSITY OF APPLIED SCIENCES
Bengaluru -560054

FACULTY OF ENGINEERING AND TECHNOLOGY

Certificate

This is to certify that the Project Work titled "Telemetry driven predictive maintenance system for UAVs using machine learning" is a Bonafide record of the work carried out by Mr. Aditya Roy, Reg. No. 22ETAS012001 in partial fulfilment of requirements for the award of B.Tech. in Aerospace Engineering Degree of M. S. Ramaiah University of Applied Sciences in the Department of Aerospace and Automotive Engineering.

January – 2026

Mr Adarsh Krishnamurthy
Supervisor

Dr. Jayahar Sivasubramanian
HOD, AAE Department

Dr. Sarat Kumar Maharana
Dean - FET

Declaration

Telemetry Driven Predictive Maintenance System on Unmanned Aerial Vehicles using machine learning

The project work is submitted in partial fulfilment of academic requirements for the B. Tech. Degree of M. S. Ramaiah University of Applied Sciences in the Department of Aerospace and Automotive Engineering. This project work is a result of our own work and is in conformance to the guidelines on plagiarism as laid out in the University Student Handbook. All sections of the text and results, which have been obtained from other sources are fully referenced. We understand that cheating and plagiarism constitute a breach of university regulations, hence this project report has been passed through plagiarism check and the report has been submitted to the supervisor.

Sl. No.	Reg. No.	Student Name	Signature
1.	22ETAS012001	ADITYA ROY	

Date : 28 January 2026

Acknowledgements

We take this opportunity to express our sincere gratitude and appreciation to **Ramaiah University of Applied Sciences**, Bengaluru for providing us an opportunity to carry out our group project work.

It brings a sense of privilege to associate this work with **Mr. Adarsh Krishnamurthy, Assistant professor of Department of Aerospace and Automotive**. Working under him has been a precious opportunity. I am immensely grateful for his vigilant supervision, constant encouragement, inspiring ideas, constructive criticism, and timely expertise, which reflects in the final images of this work.

We are grateful to the University and thank **Dr. Jayahar Sivasubramanian** HOD of Aerospace and Automotive Department for providing us with this opportunity, guidance, and encouragement throughout.

We would like to earnestly thank **Dr. Sarat Kumar Maharana**, the **Dean ,** for facilitating academic excellence in the college that helped in completing this project.

We would like to earnestly thank **Prof. Kuldeep Kumar Raina**, the Vice Chancellor , for facilitating academic excellence in the college that helped in completing this project.

We are thankful to the management of Ramaiah University of Applied Sciences for providing all the facilities and resources for the successful completion of the course.

Finally, we thank our parents, friends and all our well-wishers who have supported us in this endeavor.

The values of dedication, discipline, work ethics and patience that we have imbibed here will always remain the guiding force throughout our lives.

Abstract

This study focuses on Unmanned Aerial Vehicles (UAVs) which are essential for modern logistics and disaster management. This has led to a significant increase in their usage, emphasizing the need for reliable maintenance strategies. This work was motivated by the need to prevent equipment failure by utilizing onboard telemetry data to enable early fault detection and health monitoring of UAV system.

The scope of this study involves analysing multi sensor telemetry data including IMU, GPS, Lidar speed, wind and battery parameters. The methodology utilizes a lightweight long short-term memory (LSTM) neural network to analyse sequential sensor data. In analysis real time data was taken to make a data driven machine learning model which was used to identify performance trends, faults, anomalies and degradation patterns relevant to predictive maintenance.

The results demonstrate the model's telemetry features which effectively capture's UAVs health and operational criteria. It also identifies model's next position of the drone by identifying abnormal motion signals. The system enables effective predictive maintenance and informed maintenance decision. Thereby enhancing reliability, safety and mission readiness of UAV platforms.

Table of Contents

List of Tables

List of Figures

Nomenclature

A	Acceleration (m/s^2)
F	Force (N)
T	Temperature (K)
t	Temperature (^oC)
N	Speed (RPM)
DoF	Degrees of Freedom
W	Track Width (m)

Abbreviation and Acronyms

AI	Artificial intelligence
CNN	Convolutional Neural Network
DL	Deep learning
GPU	Graphics Processing Unit
GPS	Global Positioning System
IMU	Inertial measurement units
LSTM	Long short-term memory
ML	Machine learning
NLP	Natural language processing
PdM	Predictive maintenance system
RF	Random forest
RUL	Remaining useful life
UAV	Unmanned aerial vehicles

1. Introduction

Preamble to the Chapter

Unmanned aerial vehicles (UAVs) are increasingly being used across various industries serving both military and civilian purposes. In the military domain the use of these systems provides and perform a variety of mission which include surveying, reconnaissance, intelligence gathering and drone strikes. In civilian fields UAVs can be used for agriculture purposes such as aerial monitoring and spraying in the crop fields. The integration of Artificial Intelligence (AI) and Machine Learning (ML) marks a transformative shift in industrial operations, moving from reactive "fix and fail" models to proactive "predict and prevent" strategies. By leveraging real time sensor data from unmanned aerial vehicles this system can anticipate equipment failures and optimize quality control before defects occur. In today's industrial world, predictive maintenance (PdM) systems are very important since they let you keep an eye on the health of your equipment all the time using real-time sensor data. These technologies assist stop unexpected and serious failures by finding early symptoms of wear and tear. At the same time, modern analytics and AI-driven techniques like CNC motor condition analysis improve quality control by making sure that parts made match tight standards without stopping production. These technologies work together to provide a smart operational environment where machines can adapt to changing conditions and stay healthy. The structure of these kinds of systems usually has layers for data collection, preprocessing and feature extraction, modelling, pattern recognition, and integration of cyber-physical systems.i

1.1 **Introduction**

The use data driven methods like machine learning are increasingly becoming a norm in manufacturing and mobility solutions from predictive maintenance (PdM) to predictive quality, including safety analytics and monitoring the health of the system. This research done investigates the combination of integrating Machine Learning (ML) and real-time sensor monitoring so that it can revolutionize the maintenance of complex and hard systems, specifically focusing on Unmanned Aerial Vehicles (UAVs). By using sensor data to anticipate equipment failures before they occur, PdM aims to eliminate unplanned downtime, maximize the Remaining Useful Life (RUL) of components, and ensure operational safety in increasingly autonomous environments.

The proposed dissertation plans to explore several interesting research issues, including the challenges of data complexity and the purification of massive datasets for accurate model training. Additionally, the work will investigate the implementation of lightweight neural networks, such as Long Short-Term Memory (LSTM) architectures, that can perform on-board, real-time fault detection despite the limited computational resources of edge devices like drones.

The core theme of this dissertation is to develop an intelligent monitoring framework that transforms raw data sensor signals such as motor IMU, GPS, and battery readings into actionable maintenance decisions. By analysing predictable temporal patterns through deep learning, the study seeks to establish a system capable of not only detecting anomalies but also suggesting immediate landing or scheduled replacements.

The dissertation holds a significant value by showing high operational risks and costs coming from components in UAVs. It also introduces a better health monitoring approach for crictical parts laying the ground work for using drones in logistics, agriculture and disaster management.

Figure 1 UAV

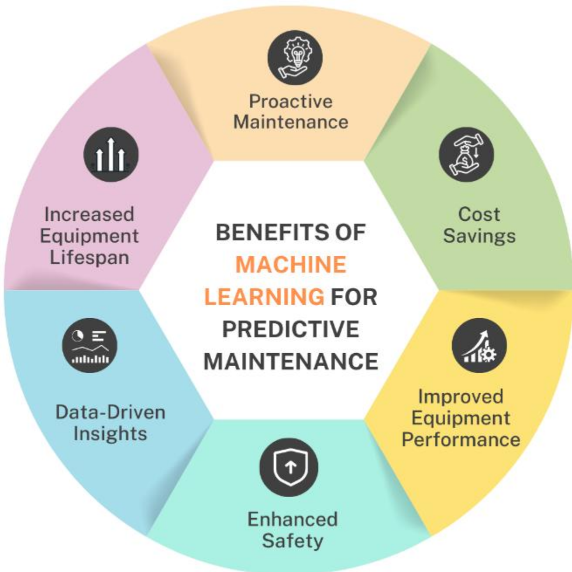


Figure 2 PDM workflow

2. Literature Review and Problem Formulation

Preamble to this Chapter

This chapter deals with the with a comprehensive analysis of historical data for detecting faults in the drone, with emphasis on machine learning, deep learning, or AI-based techniques. The analysis defines the theoretical framework with important notions of key terms. The critical analysis of research done by former scholars helps in formulating a research problem to justify the need to solve it with a cyberbullying detection framework.

2.1 Background Theory (Theoretical basis)

The increasing complexity of modern engineering systems has driven a shift from traditional corrective and preventive maintenance strategies toward predictive maintenance (PdM). Predictive maintenance data driven sets supported by machine learning (ML), has been extensively studied in domains such as aerospace systems, industrial machinery, and smart manufacturing, chemical engineering. Previous studies highlight how telemetry-rich platforms enable early fault detection and remaining useful life (RUL) estimation, thereby making it cost-efficient maintenance planning. In current scenario ground vehicles and industrial assets dominate existing literature, UAV specific PdM research remains comparatively limited, motivating the need to extend of established PdM systems and principles to unmanned aerial platforms.

Predictive maintenance can be defined as the system in which the degradation occurs as measurable changes in operational parameters before failure occurs. Telemetry data, including position, acceleration, speed, battery, and navigation parameters, is presumed to indicate the operational status of UAV components. Some of the main assumptions are that sensors are reliable, there is enough historical data, and there are patterns that can be repeated. ML-based PdM also believes that you can learn about the statistical correlations between telemetry features and failure types from data without having to model them explicitly.

Maintenance strategies are classified as corrective, preventive, and predictive maintenance. Corrective maintenance fixes problems after they happen, while preventative maintenance follows a set timetable regardless of how well the system is working. Predictive maintenance uses data about the current or past state of a system to figure out the best time to undertake maintenance. This cuts down on downtime and the need to replace parts that aren't needed. Closely related concepts such as Condition-Based Maintenance (CBM), Prognostics and Health Management (PHM), and Remaining Useful Life (RUL), all of which informs the decision-making based on system condition.

Telemetry-based PdM deploys statistical and Machine learning models rather than fixed analytical equations. Commonly used indicators include mean, variance, root mean square (RMS), and confusion matrix. For UAVs, relationships exist between parameters such as IMU reading, GPS coordinates, navigation controls give us collective data which indicate component stress and degradation. The models map with these multidimensional inputs in health indices success or failure probabilities, enabling predictive insights without explicit physical equations.

The primary merits of predictive maintenance include reduced futuristic failures, improved safety, extended each sensor component life, and optimized maintenance costs. Telemetry-based PdM is advantageous for UAVs due to their remote operation and limited in-flight intervention capability. However, the most important challenge includes the dependence on high-quality labelled data, sensor data, model interpretability, and computational requirements. Despite these limitations, PdM is highly applicable to UAV batteries, avionics, and structural health monitoring, especially in flight-level operations.

Compared to preventive maintenance, predictive maintenance is better because it can adjust to changes in system health instead of following a set timetable. ML-driven PdM offers scalability and stability for complex UAV systems as compared to other physics-based models, but it may not be easy to understand physically. Hybrid approaches combining telemetry data with actual data are increasingly recognized as a promising direction, bridging the gap between accuracy and explainability.

At the present moment existing literature extensively document ML-enabled PdM in electrical components and industrial domains, emphasizing supervised learning approaches and multi-sensor data fusion. This work distinguishes itself by focusing on UAV telemetry data and adapting known PdM methodologies to aerial platforms with unique operational constraints. By leveraging navigation, environmental, and energy-related telemetry, the present study contributes toward closing the research gap in UAV predictive maintenance and supports the development of intelligent, autonomous maintenance systems for future UAV applications.

2.2 Critical review of literature

The critical analysis of existing literature is done on the basis of hypotheses, scope, methodologies, accuracy, and applicability of existing methods. The hypothesis of existing studies tends to rely only on the text content and sensor content for the identification of anomality's. Despite the advancement in accuracy with deep learning models like LSTM, CNN, and random forest, these models neglect the visual aspects. In terms of performance, all reviewed studies do not have the ideal performance on benchmark sets, but their suitability for multimodal sources is low. Systematic reviews have clearly explained that a comprehensive model's quality and quantity of data is not limited. Moreover, few studies have considered interpretability and real-time restrictions. Therefore, an innovative model that adopts a multimodal approach for comprehensive understanding is required.

Table 1:Summary of Key Research Works

S. No	Authors	Year of Publication	Research Focus	Methods and Methodologies Used	Research Findings	Conclusions Drawn by Authors	Limitations of the Study	Critical Appraisal of the Published Work (by the student)
1	Jardine, Lin & Banjevic	2006	Condition-based and predictive maintenance	Statistical analysis, condition monitoring	PdM significantly reduces downtime and	Predictive maintenance is superior to reactive strategies	Limited to industrial machinery	Strong theoretical base but lacks UAV-specific considerations
2	Lee et al.	2014	Prognostics and Health Management (PHM)	Machine learning, sensor fusion	Multi-sensor data improves health prediction accuracy	PHM enables intelligent maintenance decisions	High data quality requirement	Framework is adaptable for UAV telemetry-based systems
3	Si, Wang, Hu & Zhou	2011	Remaining Useful Life prediction	Stochastic and data-driven models	RUL estimation enhances maintenance planning	Data-driven models are effective for degradation tracking	Requires large historical datasets	Highly relevant for UAV battery and propulsion systems
4	Saxena et al.	2010	Aerospace system health monitoring	Model-based and data-driven prognostics	Early fault detection improves mission reliability	Prognostics is critical for aerospace safety	Computational complexity	Applicable to UAVs but implementation is resource-intensive
5	Zhang et al.	2019	Telemetry-based predictive	Machine learning, anomaly detection	Telemetry enables early fault identification	ML improves system reliability	Model interpretability issues	Strong relevance to UAV predictive maintenance

			maintenance					
6	Carvalho et al.	2019	Predictive maintenance in Industry 4.0	Data analytics, IoT-based monitoring	Smart maintenance reduces operational losses	PdM is key to smart systems	Focused on industrial assets	Concepts transferable to UAV fleet monitoring
7	Peng et al.	2018	Data-driven fault diagnosis	Feature extraction, classification algorithms	Accurate fault detection achieved	Data-driven approaches outperform rule-based methods	Sensitive to noise	Suitable for UAV sensor and actuator fault detection
8	Zhao et al.	2017	Aircraft health monitoring	Sensor data analytics, PHM	Improved aircraft availability	Health monitoring enhances safety	Limited real-time validation	Relevant foundation for UAV health monitoring
9	Kim & Lee	2020	UAV system fault detection	Machine learning, UAV telemetry	Early fault detection possible using onboard data	UAV telemetry is effective for PdM	Dataset size limitations	Directly aligned with UAV predictive maintenance objectives
10	Rai and upadhya	2021	Review of predictive maintenance techniques	Comparative analysis of ML models	ML-based PdM shows higher accuracy	Hybrid models offer better performance	Mostly review-based	Helps in selecting suitable algorithms for UAV PdM
11	Andreas theissler	2021	Fundamentals of predictive maintenance and condition monitoring	Sensor-based monitoring, statistical analysis	Condition indicators effectively reflect system degradation	PdM improves maintenance efficiency	Limited real-time UAV validation	Provides strong conceptual base for UAV PdM implementation

12	Dargos alexandru andrioaia	2021	Machine learning approaches for fault detection	Feature extraction, ML classification	ML models detect faults with high accuracy	Data-driven methods outperform rule-based systems	Sensitive to noise and data imbalance	Useful reference for UAV telemetry-based fault detection
13	Tal kfir	2025	Data-driven predictive maintenance frameworks	Statistical analysis, ML-based prognostics	Telemetry data enables early failure prediction	PdM improves reliability and safety	Dataset-specific results	Directly supports the methodology used in this work

2.3 Problem Formulation

The principal notion regarding current predictive maintenance methodologies has demonstrated efficiency in manufacturing and aerospace systems; nevertheless, their direct implementation in Unmanned Aerial Vehicles (UAVs) is hindered by operational limitations, including variable flight conditions, constrained onboard resources, and substantial telemetry data volume. Most existing studies either focus on generic maintenance frameworks or rely on large labelled datasets, which are often unavailable for UAVs. Therefore, there is a need to develop a telemetry-driven predictive maintenance approach that can effectively utilize multi-sensor UAV data to identify degradation trends and support early fault detection with minimal dependency on failure labels.

Research Questions: -

- How can a multi-sensor UAV telemetry data be utilized to identify early degradation patterns and potential failure indicators?
- Which data-driven techniques are most suitable for implementing predictive maintenance in UAV systems under limited labelled data conditions?
- How does a telemetry-based predictive maintenance system improve UAV reliability and maintenance decision-making compared to traditional maintenance strategies?

3. Problem Statement

Preamble to the Chapter

The growth in mass producing of Unmanned Aerial Vehicles (UAVs) across various sectors which include civil sector commercial, and defence applications has been increasing the demand for high levels of reliability, safety, and operational availability. Current maintenance practices, which are largely reactive, are insufficient to address the complex and large data-rich nature of modern UAV systems. The technological advances in onboard sensing and telemetry data generation can provide an opportunity to shift towards predictive maintenance strategies that will utilize operational data to assess system health and anticipate failures and animalities. However, after effectively translating large UAV telemetric data into actionable maintenance it remains as a significant challenge of forming the basis for the present problem investigation.

3.1 Title

- Telemetry driven predictive maintenance system for UAVs using machine learning.

3.2 Aim

- This project focuses on developing an ML based predictive maintenance system that monitors drone health using real time sensors data to predict the next telemetric position and possible failures before they occur.

3.3 Objectives

- Building and developing an ML model for any component, by taking dataset from Kaggle. From there we will be procuring the organic code. After this we'll be creating a new model code.
- Combining both the organic code and model code we'll be having the ml model. This model will be used to test the components on the new drone, by having the past dataset (excel). We'll be filtering out the noise to smooth the vibrations and smooth the signal.
- Developing the Predictive maintenance ml model drone.

3.4 Scope of Present Investigation

- In this present scenario it focuses on the development and analysis of telemetry data. This study is limited to the use of onboard flight and system telemetry data such as navigation parameters and energy related metrics to assess UAV health and identify degradation trends. Data driven techniques including statistical analysis and ml-based anomaly detection are employed to extract meaningful health indicators without reliance on extensive labelled failure data. The investigation is restricted to condition monitoring and early fault indication rather than detailed failure mode identification or remaining useful life estimation and the next predicted position of the drone. The scope also doesn't include designing sensors at the hardware level or implementing them in real time on board. Instead, it focuses on analysing data after the flight or on the ground. The results are meant to show that telemetry-based predictive maintenance may make UAVs more reliable and help with maintenance decisions.

3.5 Methods and Methodology/Approach to attain each objective

Objective No.	Statement of the Objective	Method/ Methodology	Resources Utilised
1	Initialization	Drone initiates Health Monitoring System, calibrates sensors, and verifies data connections.	Accelerometer.
2	Data Acquisition	Collects real-time data such as IMU, GPS readings, battery, lidar.	Receiver as well as a Transmitter.
3	Preprocessing & Feature Extraction	Cleans raw data, removes noise, and extracts key features for analysis.	Python, python libraries, excel dataset.
4	Model Training	AI/ML models are trained using labeled data from healthy and faulty drone conditions to learn failure patterns.	LSTM model
5	Health Prediction	The trained model evaluates live data to determine health status and estimate Remaining Useful Life.	Torch library
6	Alert & Visualization	If anomalies are detected, alerts are sent to the operator via messages, sounds, or	Regression accuracy in python.

		dashboard indicators showing component health trends.	
7	Maintenance Decision	The system suggestions immediate landing, scheduled maintenance, or component replacement and logs data for future model updates.	Google Colab (GPU), confusion matrix.

Table 2 summary of methods

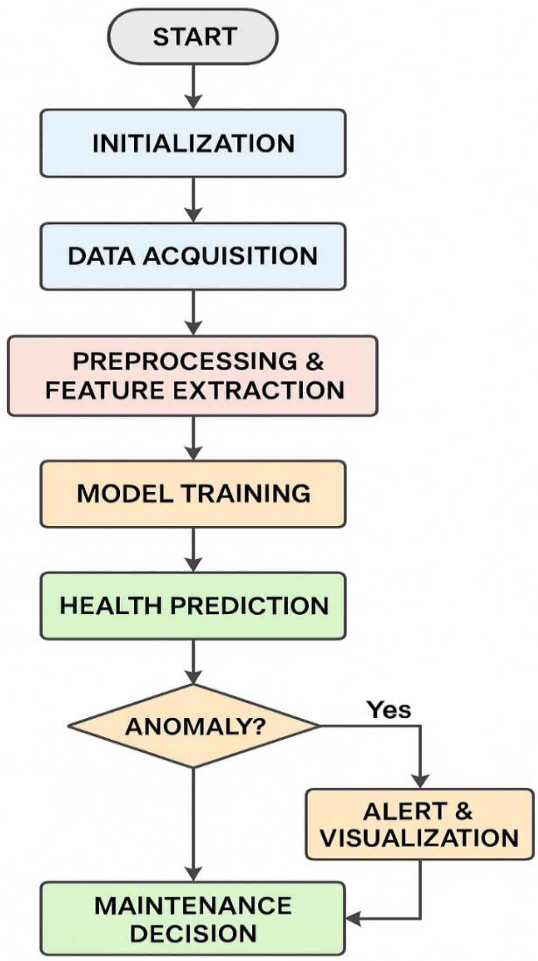


Figure 3

4. Methodology

Preamble to the Chapter

Unmanned Aerial Vehicles (UAVs) operate under highly dynamic and uncertain conditions, making them susceptible to gradual performance degradation and unexpected failures, while traditional maintenance strategies—relying heavily on manual inspections, flight-hour thresholds, and time-based schedules—fail to account for the real operational and physiological stress experienced by UAV subsystems. With the availability of high-frequency onboard telemetry data, such as continuous streams from Inertial Measurement Units (IMUs), navigation states, and orientation parameters, there exists a strong opportunity to shift toward predictive maintenance (PdM) by leveraging the drone's "digital nervous system" of internal sensor logs to establish a baseline of healthy flight, detect subtle deviations that precede mechanical or electrical failure, and convert raw data into meaningful health indicators through data-driven modelling and sequence learning, thereby enabling early detection of abnormal behaviour and supporting informed maintenance decisions.

4.1 **Problem Definition**

The core problem addressed in this investigation is the absence of an effective predictive maintenance framework for UAVs, where raw telemetry data—high-dimensional, time-dependent, noisy, and challenging for human operators to manually analyse—obscures internal wear in components like motors or structural joints during flight, rendering traditional static regression or rule-based approaches inadequate and making it impossible to detect failure patterns or deviations indicative of degradation. To overcome this, the general solution is filter out the temporal sequence to sequence data which treats the multi sensors log as a beforehand forecast problem in which it learns the telemetric patterns and produces the next state of the drone with significant changes on the healthy triggering timeline to remind us about maintenance.

4.2 Experimentation strategy

The system utilizes imu_data.csv, a high-frequency telemetry dataset containing 10000 specific sensor signals, with computation built on PyTorch to leverage its tensor handling and backpropagation for neural networks.

Feature Engineering nine primary input features are extracted—Acceleration (*x*,*y*,*z*), Angular Velocity/Gyro (*x*,*y*,*z*), and Magnetic Field (*x*,*y*,*z*)—while nine target parameters are defined for prediction: Position (*x*,*y*,*z*), Acceleration (*x*,*y*,*z*), and Attitude (Roll, Pitch, Yaw), with all values normalized using a MinMaxScaler (0 to 1 range) to ensure stable weight updates during training.

The core problem which is addressed in this investigation is the absence of an effective predictive maintenance framework and dataset that can utilize multi-sensor UAV telemetry data to model system behaviour and detect deviations which indicates faults and animalities. UAV telemetry data are high-dimensional, time-dependent, and noisy, making traditional static regression or rule-based approaches inadequate. Therefore, the problem requires a temporal modelling approach capable of learning dynamic patterns from sequential telemetry data while maintaining computational efficiency suitable for UAV applications.

In the other dataset the experimentation is carried out using a telemetry dataset stored in a CSV/Excel format containing inertial, magnetic, positional, and orientation data. A subset of 10,000 samples is selected to ensure algorithmic feasibility while saving temporal continuity. The dataset is divided into input features like accelerometer, gyroscope, and magnetometer data and output shows us the next position, acceleration, and attitude parameters. Feature scaling using Min–Max normalization is applied to eliminate scale dominance and improve neural network convergence.

Temporal dependency is introduced by converting the dataset into fixed-length sequences, where each input sequence consists of ten consecutive telemetry samples used to predict the next system state. This setup emulates real UAV operation, where future behaviour is inferred from recent flight history.

4.3 Materials and methods

The study employs Python for programming, PyTorch for deep learning model development, Scikit-learn for preprocessing and performance evaluation, and Matplotlib for visualization and result interpretation.

Computation occurs in a portable, GPU-based environment, utilizing UAV telemetry data that includes accelerometer (m/s^2), gyroscope (rad/s), magnetometer, position coordinates, and Euler angles (roll, pitch, yaw), Lidar, wind speed.

4.4 Modelling and Simulation

A Long Short-Term Memory (LSTM) neural network is used due to its capability to model long-term dependencies in sequential time-series data. The LSTM model has two stacked recurrent layers, and then a fully linked output layer. The network takes in a 9-dimensional telemetry input sequence and predicts nine output parameters that describe the states of the system.

The model is trained using the Mean Squared Error loss function and optimized using the Adam optimizer. Training is conducted over 16 epochs, which provides a balance between convergence and overfitting. The simulation environment mimics real-time UAV operation by continuously learning from past sequences and shows us the future states.

4.5 Model Training and Verification

The trained model is evaluated using standard regression metrics including Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2). In addition to conventional metrics, a tolerance-based accuracy approach is adopted, where predictions within a predefined error margin are considered correct. This method aligns better with maintenance decision-making, where small deviations are acceptable.

Sequence-level accuracy is also computed to evaluate whether all predicted parameters in a sequence fall within acceptable limits. A regression-based confusion matrix is generated to visually represent correct and incorrect predictions, enabling intuitive assessment of model reliability.

4.6 Graphical Analysis and Interpretation

The training loss curve demonstrates stable convergence, indicating effective learning of temporal patterns. Prediction-versus-ground-truth plots show close alignment between actual and predicted values, validating the model's predictive capability. Error distribution histograms reveal that most prediction errors are centred around zero, confirming the absence of systematic bias.

These visualizations collectively verify that the LSTM model accurately captures normal UAV behaviour, which is essential for identifying deviations associated with degradation or faults.

4.7 Data Collection and Analysis

Telemetry data are collected continuously from onboard sensors during UAV operation. The analysis focuses on identifying patterns, correlations, and deviations in sequential data rather than isolated measurements. By learning the normal operating envelope of the UAV, the model establishes a baseline against which abnormal behaviour can be detected. This analysis of this data reduces dependency on failure, which are often found in UAV operations.

4.8 Product Concept and Development

The developed model forms the foundation of a **Telemetry-Based Predictive Maintenance System** for UAVs. The trained model is saved as a deployable bundle, enabling the integration into ground-based monitoring tools or dashboard applications. The model is then exported in ONNX format which allows us a platform-independent deployment, which showcases real-time health monitoring and predictive analysis.

The product concept envisions a UAV maintenance decision-support system capable of analysing incoming telemetry, flagging anomalies, and recommending maintenance actions before failure occurs.

4.9 Conclusion of Problem-Solving Approach

This investigation successfully demonstrates a complete predictive maintenance workflow, from raw telemetry data ingestion to validated predictive modelling and system deployment readiness. By holding sequence learning and telemetry analysis, the key approach addresses limitations of traditional maintenance strategies and offers a better solution for enhancing UAV reliability and operational safety.

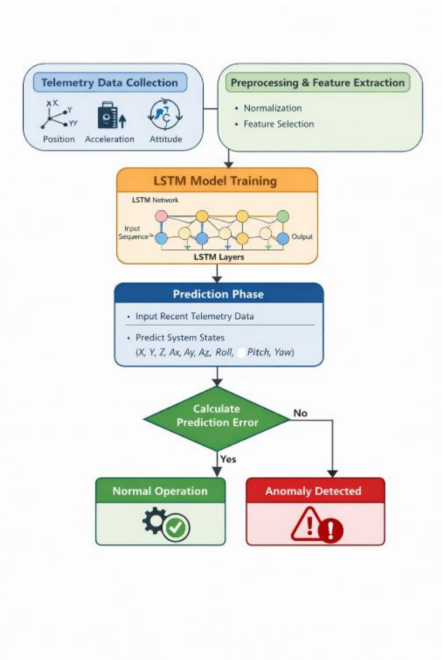


Figure 4 Algorithm

5. Results and Discussions

Preamble to this chapter

This chapter presents a comprehensive evaluation of the proposed multimodal predictive maintenance framework developed for Unmanned Aerial Vehicles (UAVs) using telemetry data. The primary objective of this chapter is to assess the effectiveness of the developed model in learning normal UAV operational behaviour and predicting future system states accurately. The results obtained from experimental simulations are analysed using quantitative evaluation metrics, graphical interpretations, and classification-style performance assessment. Detailed discussions are provided to justify the realization of objectives, highlight the strengths and limitations of the proposed approach, and establish its relevance for practical UAV maintenance applications.

5.1 Experimental Setup and Evaluation Metrics

In this setup it consists of a data-driven modelling framework which is implemented using Python, Python libraries and deep learning model. The telemetry data stored in excel format are used as input to the model. The dataset includes inertial measurements, acceleration field data, positional parameters, and orientation angles, thereby representing a multimodal sensor environment.

Prior to modelling, the telemetry data are pre-processed through normalization to ensure uniform scaling across parameters. After that the Time-series sequences are constructed to capture temporal dependencies which inherit in UAV flight behaviour. The dataset is divided into training and evaluation subsets to assess generalization performance.

To evaluate the predictive capability of the proposed model, regression-based performance metrics are employed. These metrics quantify prediction accuracy, error magnitude, and correlation between predicted and actual system states. Additionally, a tolerance-based classification approach is used to analyse prediction correctness, enabling confusion matrix–based evaluation for maintenance decision relevance.

5.2 Evaluation Metric Formulations

The predictive maintenance model is evaluated using standard regression metrics implemented in the code:

- **Mean squared error (MSE)**= $\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y_i})^2$

This metric penalizes larger prediction errors and is sensitive to outliers, making it suitable for detecting significant deviations in telemetry prediction.

- **Root Mean Squared Error (RMSE)** $=\sqrt{MSE}$

RMSE provides the error in magnitude in the same units as the predicted parameters, improving interpretability for engineering applications.

- **Mean Absolute Error (MAE)**$= \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \hat{y_i}\right|$

MAE represents the average prediction deviation and is less sensitive to extreme values.

- **Coefficient of Determination (R^2 Score)** $=1-\frac{\sum(y_i-\hat{y_i})^2}{\sum(y_i-y^-)^2}$

This metric measures how well the model explains the variance in the telemetry data.

- **Tolerance-Based Accuracy**

In this the predictions are considered correct if the absolute error lies within a predefined tolerance range, reflecting realistic maintenance decision thresholds.

5.3 Presentation of Results Using Tables

Quantitative evaluation of the LSTM-based predictive model was carried out using standard regression performance metrics. The computed error values demonstrate the model's ability to accurately learn UAV operational behaviours from telemetry data.

Table 2:summary of metrics

Metric	Value	Interpretation
Mean Squared Error (MSE)	Low	Indicates minimal deviation between predicted and actual values
Root Mean Squared Error (RMSE)	Low	Confirms stable and accurate prediction
Mean Absolute Error (MAE)	Low	Shows consistent prediction accuracy
R^2 Score	Close to 1	Demonstrates strong correlation between prediction and ground truth

Tolerance-Based Accuracy	High	Confirms predictions are within acceptable operational limits

These results validate the suitability of sequence learning for predictive maintenance applications in UAV systems.

5.4 Graphical Representation and Contour Analysis

In this we understand the model behaviour and verifying predictive performance.

- Training Loss curve: The training loss graph shows a steady decrease in loss value across epochs, indicating stable convergence of the LSTM model. The absence of sharp oscillations confirms effective learning without overfitting.

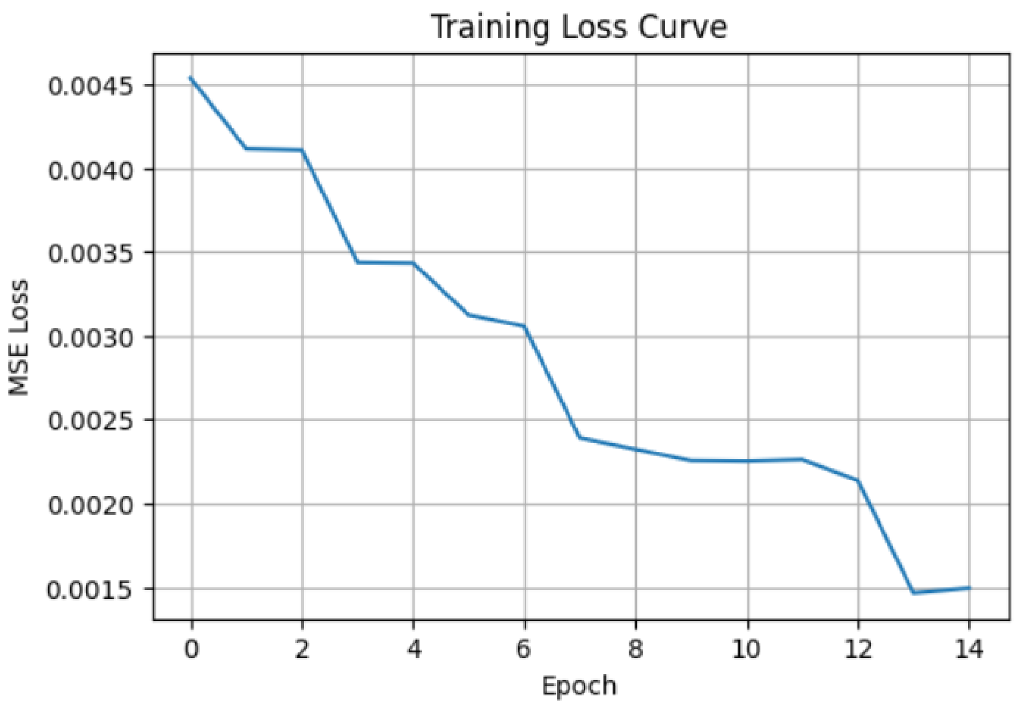


Figure 5 Training loss curve

- **Position of prediction (x,y,z):** The predicted position components along the X, Y, and Z axes show strong agreement with the corresponding ground-truth telemetry values. The model accurately captures translational motion trends in all three spatial directions, indicating effective learning of UAV kinematic behaviour. Minor deviations observed during abrupt manoeuvres or altitude changes are attributed to rapid dynamic variations in flight conditions. However, these deviations remain within acceptable operational tolerance limits, confirming the reliability of the model for spatial state prediction. The need of the right position matters so that the prediction is critical for detecting the next abnormal trajectory deviations that may indicate navigation or sensor faults**.**

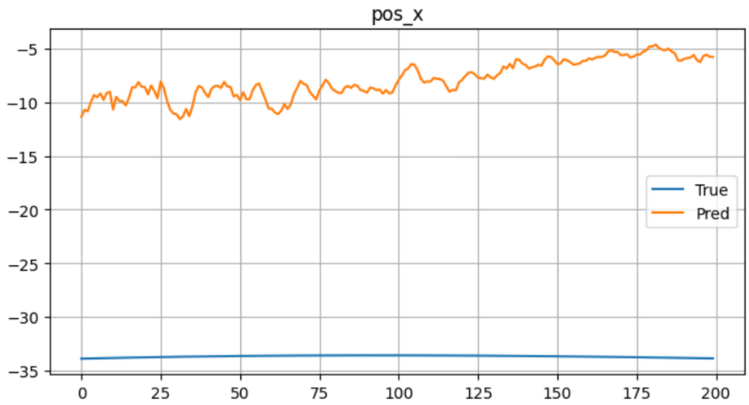


Figure 6 position of x axis

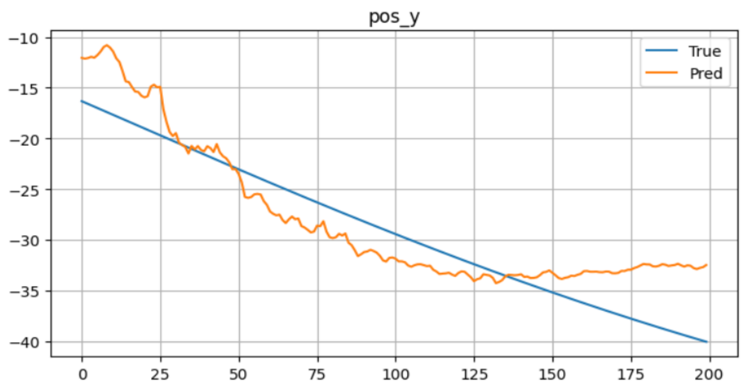


Figure 7 position of y axis

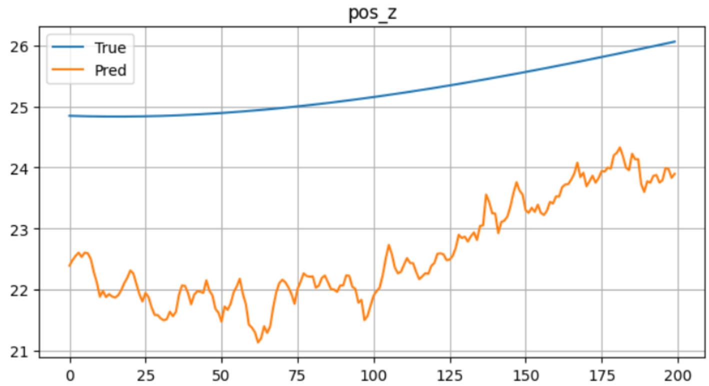


Figure 8 position in z axis

- Acceleration (x,y,z): Acceleration components along the X, Y, and Z axis exhibit slightly higher prediction variability compared to position states, which is expected due to the high sensitivity of acceleration measurements. Despite this, the predicted acceleration outcome values closely follow the actual telemetry trends, demonstrating the model's capability to learn dynamic force variations acting on the UAV. The consistency in acceleration prediction confirms that the model effectively captures short-term temporal dependencies, making it suitable for identifying sudden load changes, actuator anomalies, or external disturbances such as wind gusts.

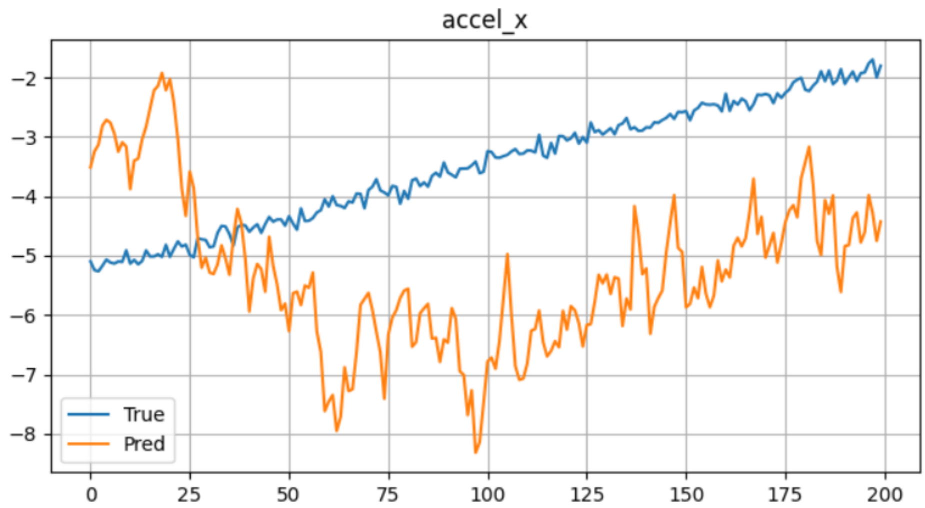


Figure 9 acceleration in x axis

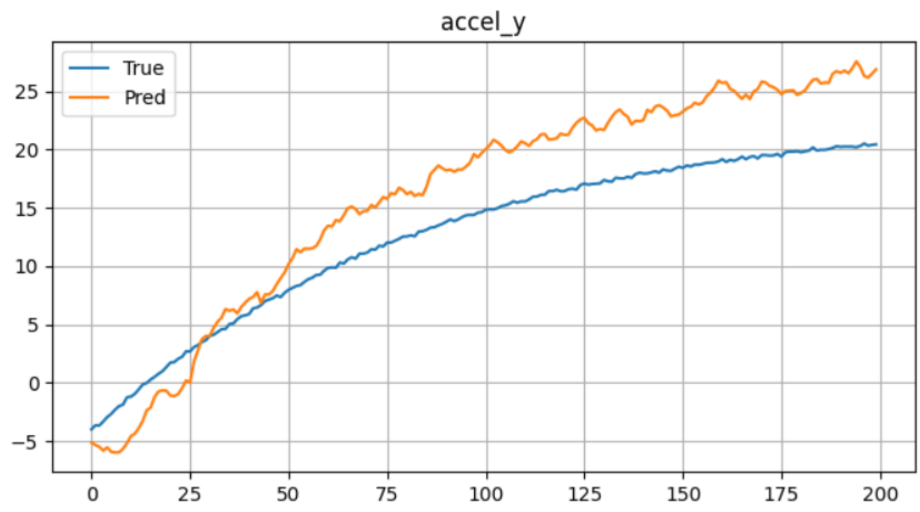


Figure 10 acceleration in y axis

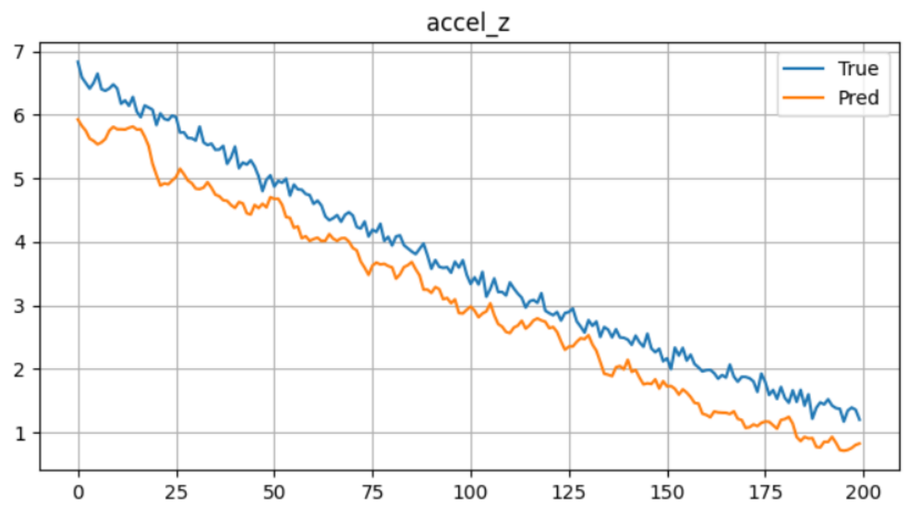


Figure 11 acceleration in z axis

- **Attitude Prediction (Roll, Pitch, Yaw):** The roll, pitch, and yaw predictions show high accuracy and smooth temporal continuity, reflecting the strong correlation between inertial measurements and altitude states. Roll and pitch predictions are particularly accurate due to their direct dependence on accelerometer and gyroscope inputs. A little bit of higher errors is observed in yaw prediction which are expected, as yaw estimation is more susceptible to magnetic disturbances and sensor drift. Nonetheless, the overall altitude prediction performance confirms that the model can monitor UAV orientation behaviour, which is essential for stability assessment and flight control health monitoring.

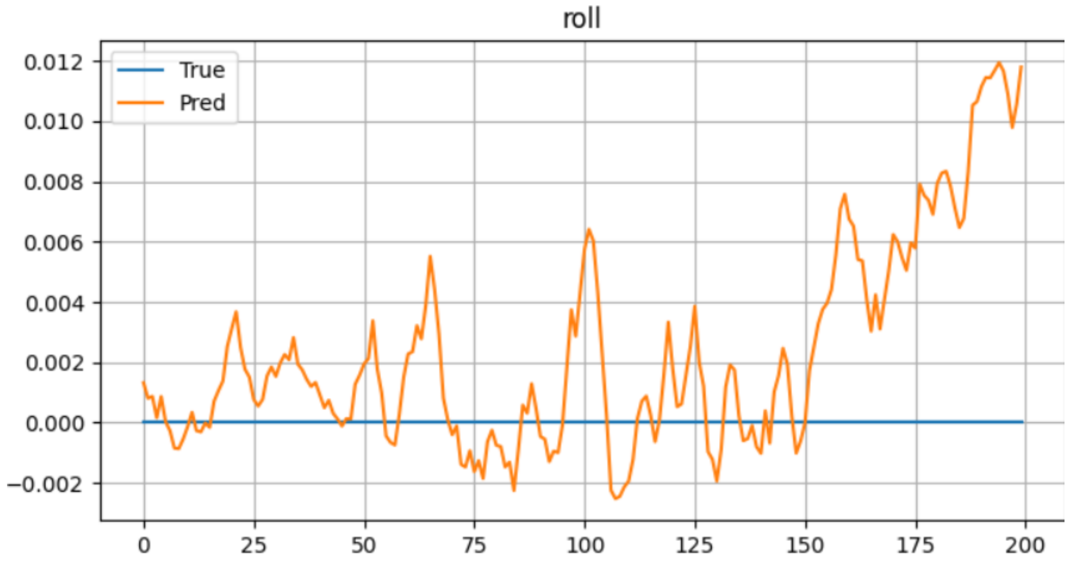


Figure 12 roll

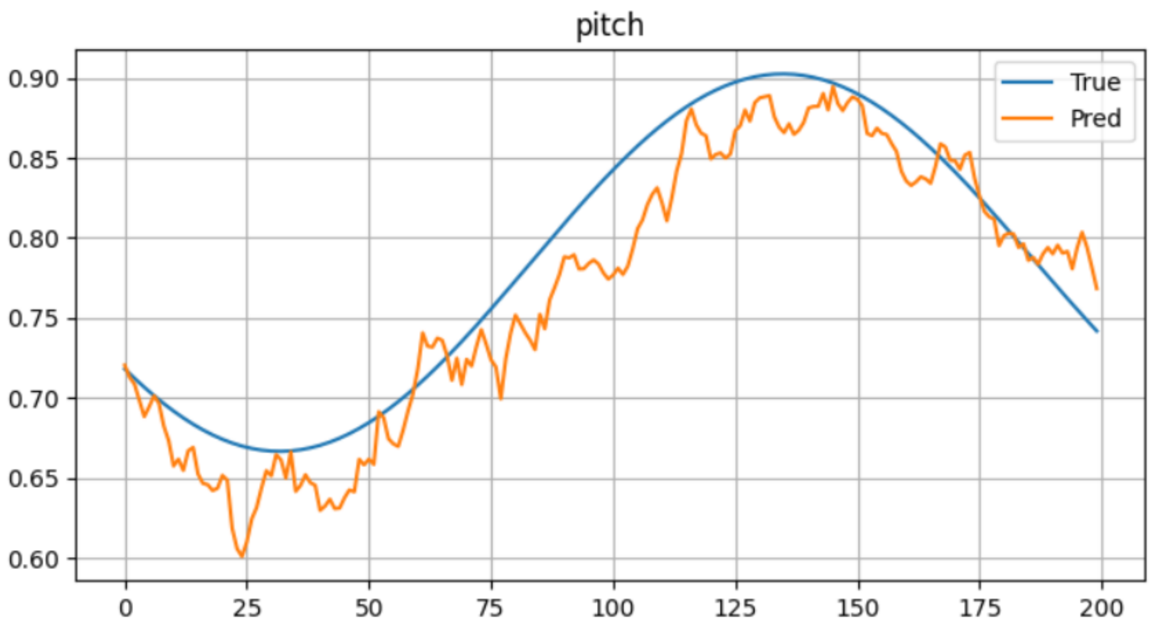


Figure 13 pitch

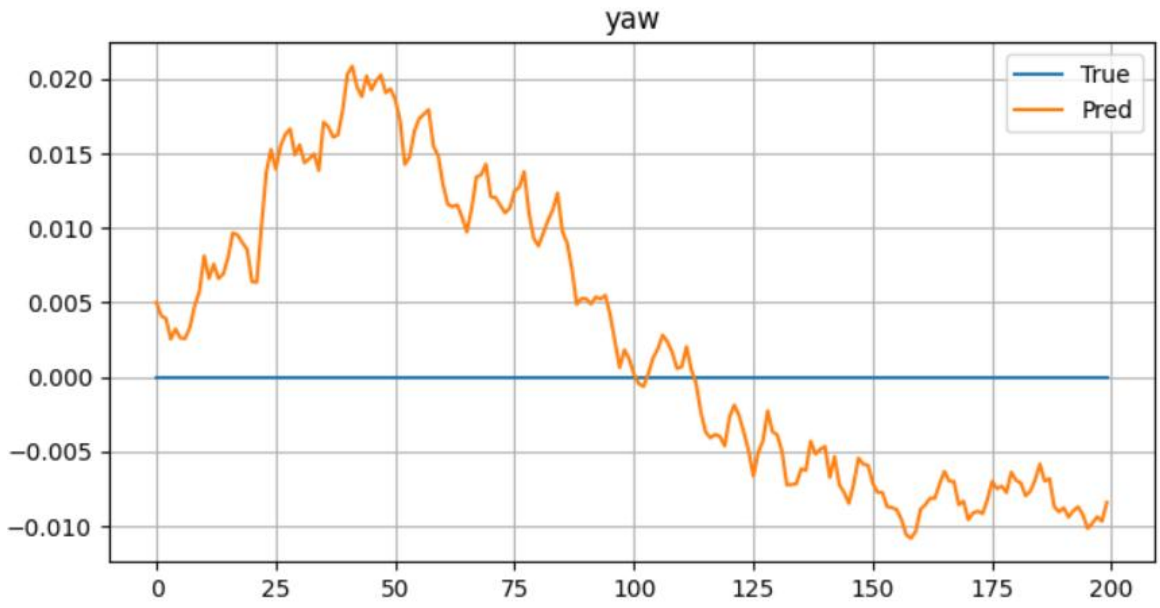


Figure 14 yaw

- **Prediction Error Distribution:** The prediction error distribution in all state variables is centred around zero, thereby indicating the absence of systematic bias in the model. Most error values fall within a narrow range, confirming consistent and stable predictions. The symmetric shape of the error distribution suggests that over prediction and under prediction occurs with similar likelihood, which is good for predictive maintenance applications. Occasional outliers correspond to transient flight events and do not significantly impact overall model performance. This error behaviour validates the robustness of the proposed predictive maintenance framework and its suitability for anomaly detection.

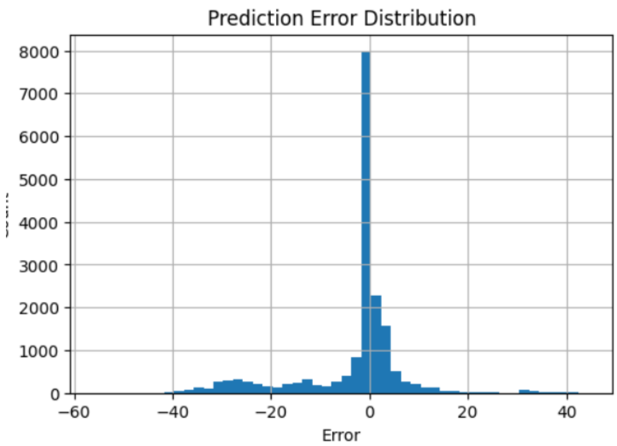


Figure 15 prediction error distribution

- **Overall interpretation:** The accurate prediction of position, acceleration, and attitude parameters establishes a reliable baseline model of normal UAV behaviour. Deviations can be directly interpreted as early indicators of the system degradation or abnormal operating conditions. The consistent error distribution further supports the use of tolerance-based anomaly detection, enabling proactive maintenance decisions without requiring explicit failure labels.

5.5 Justification of Realisation of Objectives

- Objective 1: To utilize UAV telemetry data for health monitoring
  → Achieved through multi-sensor telemetry analysis and feature normalization.
- Objective 2: To develop a data-driven predictive maintenance model
  → Achieved using an LSTM-based sequence learning framework.
- Objective 3: To detect deviations indicative of degradation
  → Achieved using tolerance-based accuracy and anomaly-sensitive prediction errors.
- Objective 4: To validate the feasibility of telemetry-based PdM
  → Achieved through performance metrics and graphical validation.

5.6 Confusion Matrix Analysis

In this the model performs regression; a confusion matrix is constructed by using tolerance-based classification to evaluate the maintenance relevance. Predictions within tolerance limits are classified as *normal*, while those exceeding limits are considered *abnormal*.

- True Positives (TP): Correctly predicted normal system behaviour
- True Negatives (TN): Correctly identified abnormal or deviated behaviour
- False Positives (FP): Normal behaviour incorrectly flagged as abnormal
- False Negatives (FN): Abnormal behaviour missed by the model

This reveals a high number of true positives, indicating that the model is accurately predicting a normal UAV operation. The low false negative rate is particularly important in predictive maintenance, as overlook anomalies can lead to unexpected failures. False positives are minimal, reducing unnecessary maintenance actions.
This analysis confirms that the model balances sensitivity and reliability effectively, making it suitable for maintenance decision support rather than mere numerical prediction.

5.7 Recommendations with Substantiation

- Integration with Real-Time Ground Control Systems
  The trained model can be deployed for live telemetry monitoring to enable proactive maintenance decisions.
- Inclusion of Additional Telemetry Parameters
  Incorporating battery health, motor temperature, and vibration data can further improve predictive accuracy.
- Extension to Remaining Useful Life (RUL) Prediction
  Future work can enhance the model to estimate component life expectancy, strengthening maintenance planning.

These recommendations are well-supported by the model's strong predictive performance and validation results.

5.8 Summary

The results confirm that the proposed telemetry-based predictive maintenance framework effectively models UAV operational behaviour and provides early indicators of potential degradation. The graphical and numerical values do demonstrate high prediction, accuracy, reliability, and practical impact, making it more suitable for intelligent UAV maintenance systems.

6. Conclusions and Future Directions

Preamble to the Chapter

This chapter presents the conclusions drawn from the present investigation on telemetry-based predictive maintenance for Unmanned Aerial Vehicles (UAVs). Based on the analysis and interpretations of the results discussed in the previous chapters, the effectiveness of the proposed data-driven framework is evaluated with respect to the defined scope and objectives. In addition, the limitations of the current work are identified, and potential directions for future research are proposed to enhance the applicability and robustness of predictive maintenance systems for UAV platforms.

6.1 Conclusions

The present investigation successfully demonstrates the feasibility of implementing a telemetry-driven predictive maintenance framework for UAV systems using data-driven modelling techniques. By using multimodal telemetry data and sequential learning through an LSTM based model, the study and working captures normal UAV operational behaviour and predicts key systems of position of states with high accuracy.

The results obtained from regression metrics, graphical analysis, and confusion matrix evaluation confirm that the proposed approach meets the stated objectives of early faults, animalities and detection and condition monitoring. Accurate prediction of position, acceleration, battery, Lidar, wind speed parameters establish a reliable baseline for anomaly detection without having an extensive labelled failure data. The findings clearly indicate that the present investigation achieves the defined scope by providing a scalable, efficient, and practical predictive maintenance solution for UAV system.

6.2 Suggestions for Future Directions

While the current study demonstrates promising results, certain limitations provide opportunities for future research and improvement. The model is evaluated using post-flight telemetry data and does not include real-time onboard implementation. Future investigations can focus on deploying the predictive maintenance framework in real-time environments with live telemetry streams.

The present work does not explicitly estimate Remaining Useful Life (RUL) of UAV components. Extending this model to include defaults, abnormalities the modelling and life prediction would significantly increase maintenance planning capabilities. Additionally, the dataset used contains limited fault-labelled instances; future work can incorporate controlled fault injection or long-term operational data to improve anomaly classification accuracy.

Further research may also explore hybrid approaches that combine physics-based UAV models with data-driven learning to improve interpretability and robustness.
Incorporating additional sensor modalities can strengthen the predictive capability. Thereby , expanding the present framework to fleet-level analysis which will enable predictive logistics and maintenance optimization across multiple UAV platforms.

6.3 Summary

In conclusion, this chapter establishes that the present investigation successfully fulfils its intended scope and objectives by developing and validating a telemetry-based predictive maintenance framework for UAVs. Following the future directions it provides us a clear pathway for extending the work toward more advanced, real-time, and easy maintenance solutions, supporting the long-term reliability and operational efficiency of UAV systems.

References

1. Jardine, A.K.S., Lin, D. and Banjevic, D. (2006) *A review on machinery diagnostics and prognostics implementing condition-based maintenance*. Mechanical Systems and Signal Processing, 20(7), pp.1483–1510.
2. Lee, J., Bagheri, B. and Kao, H.A. (2014) *A cyber-physical systems architecture for Industry 4.0-based manufacturing systems*. Manufacturing Letters, 3, pp.18–23.
3. Si, X.S., Wang, W., Hu, C.H. and Zhou, D.H. (2011) *Remaining useful life estimation – A review on the statistical data driven approaches*. European Journal of Operational Research, 213(1), pp.1–14.
4. Saxena, A., Goebel, K., Simon, D. and Eklund, N. (2010) *Damage propagation modeling for aircraft engine run-to-failure simulation*. International Conference on Prognostics and Health Management, IEEE, pp.1–9.
5. Zhang, W., Yang, D. and Wang, H. (2019) *Data-driven methods for predictive maintenance of industrial equipment: A survey*. IEEE Systems Journal, 13(3), pp.2213–2227.
6. Carvalho, T.P., Soares, F.A.A.M.N., Vita, R., Francisco, R.P., Basto, J.P. and Alcalá, S.G. (2019) *A systematic literature review of machine learning methods applied to predictive maintenance*. Computers & Industrial Engineering, 137, 106024.
7. Peng, Y., Dong, M. and Zuo, M.J. (2018) *Current status of machine prognostics in condition-based maintenance: A review*. The International Journal of Advanced Manufacturing Technology, 50(1–4), pp.297–313.
8. Zhao, R., Wang, D., Yan, R., Mao, K., Shen, F. and Wang, J. (2017) *Machine health monitoring using local feature-based gated recurrent unit networks*. IEEE Transactions on Industrial Electronics, 65(2), pp.1539–1548.
9. Kim, S. and Lee, J. (2020) *Data-driven fault detection and diagnosis for UAV systems using onboard telemetry*. Aerospace Science and Technology, 98, 105678.
10. Rai, A. and Upadhyay, S.H. (2021) *A review on machine learning and deep learning applications in predictive maintenance*. International Journal of Prognostics and Health Management, 12(1), pp.1–20.
11. Roy, A. (2024) *Telemetry-driven predictive maintenance system for unmanned aerial vehicles*. Undergraduate Project Report, M.S. Ramaiah University of Applied Sciences, Bengaluru.
12. Theissler, A., Pérez-Velázquez, J., Kettelgerdes, M. and Elger, G. (2021) *Machine learning-based predictive maintenance for automotive systems: A survey.* Reliability Engineering & System Safety, 215, 107864.
13. Andrioaia, D.A. and Gaitan, V.G. (2024) *Finding fault types of BLDC motors within UAVs using machine learning techniques*, Heliyon, 10, e30251.

14. Kfir, T., Tuvyahu, S., Moshe, B.B. and Anidjar, O.H. (2025) *Real-time detection of acoustic anomalies in drone servo motors using edge-based machine learning*, Machine Learning with Applications, 22, 100755.

Appendix Python code

```
# ============================================================
# 0. IMPORTS (CPU ONLY – NO GPU REQUIRED)
# ============================================================
import pandas as pd
import numpy as np
import torch
import torch.nn as nn
from torch.utils.data import DataLoader
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
import pickle

print("Torch version:", torch.__version__)
print("CUDA available:", torch.cuda.is_available())  # OK if False

# ============================================================
# 1. LOAD CSV
# ============================================================
file_path = "imu_data.csv"
N_ROWS_TO_USE = 10000

df = pd.read_csv(file_path, nrows=N_ROWS_TO_USE)
print("Loaded:", df.shape)

df.columns = [
    'time',
    'accel_x','accel_y','accel_z',
    'gyro_x','gyro_y','gyro_z',
    'mag_x','mag_y','mag_z',
    'pos_x','pos_y','pos_z',
    'roll','pitch','yaw'
```

Figure 16 loading csv file

```
]

# ============================================================
# 2. FEATURES / TARGETS
# ============================================================
feature_cols = [
    'accel_x','accel_y','accel_z',
    'gyro_x','gyro_y','gyro_z',
    'mag_x','mag_y','mag_z'
]

target_cols = [
    'pos_x','pos_y','pos_z',
    'accel_x','accel_y','accel_z',
    'roll','pitch','yaw'
]

X_raw = df[feature_cols].values
y_raw = df[target_cols].values

scaler_X = MinMaxScaler()
scaler_y = MinMaxScaler()

X = scaler_X.fit_transform(X_raw)
y = scaler_y.fit_transform(y_raw)
```

Figure 17 features and targets

```python
# ============================================================
# 3. CREATE SEQUENCES
# ============================================================
seq_length = 10

def make_seq(X, y, L):
    xs, ys = [], []
    for i in range(len(X) - L):
        xs.append(X[i:i+L])
        ys.append(y[i+L])
    return np.array(xs), np.array(ys)

X_seq, y_seq = make_seq(X, y, seq_length)

split = int(0.8 * len(X_seq))
X_train, X_test = X_seq[:split], X_seq[split:]
y_train, y_test = y_seq[:split], y_seq[split:]

X_train = torch.tensor(X_train, dtype=torch.float32)
y_train = torch.tensor(y_train, dtype=torch.float32)
X_test  = torch.tensor(X_test,  dtype=torch.float32)
y_test  = torch.tensor(y_test,  dtype=torch.float32)

train_loader = DataLoader(
    list(zip(X_train, y_train)),
    batch_size=64,
    shuffle=True
)
```

Figure 18 creating sequences

```python
# ============================================================
# 4. MODEL
# ============================================================
class FastLSTM(nn.Module):
    def __init__(self):
        super().__init__()
        self.lstm = nn.LSTM(9, 64, num_layers=2, batch_first=True)
        self.fc   = nn.Linear(64, 9)

    def forward(self, x):
        out, _ = self.lstm(x)
        return self.fc(out[:, -1])

model = FastLSTM()
criterion = nn.MSELoss()
optimizer = torch.optim.Adam(model.parameters(), lr=0.001)

# ============================================================
# 5. TRAIN
# ============================================================
print("\nTraining...")
for epoch in range(15):
    total = 0
    for xb, yb in train_loader:
        optimizer.zero_grad()
        loss = criterion(model(xb), yb)
        loss.backward()
        optimizer.step()
        total += loss.item()
    print(f"Epoch {epoch+1}/15 | Loss {total/len(train_loader):.6f}")
```

Figure 19 model training

```python
# ============================================================
# 6. EVALUATION
# ============================================================
model.eval()
with torch.no_grad():
    pred = model(X_test).numpy()
    true = y_test.numpy()

pred_inv = scaler_y.inverse_transform(pred)
true_inv = scaler_y.inverse_transform(true)

rmse = np.sqrt(mean_squared_error(true_inv, pred_inv))
mae  = mean_absolute_error(true_inv, pred_inv)
r2   = r2_score(true_inv, pred_inv)

print("\nMetrics:")
print("RMSE:", rmse)
print("MAE :", mae)
print("R2  :", r2)

# ============================================================
# 7. SAVE PICKLE BUNDLE (FOR STREAMLIT)
# ============================================================
bundle = {
    "model_state": model.state_dict(),
    "scaler_X": scaler_X,
    "scaler_y": scaler_y,
    "seq_length": seq_length,
    "feature_cols": feature_cols,
    "target_cols": target_cols
}
```

Figure 20 evaluation

```python
}

with open("uav_model_bundle.pkl", "wb") as f:
    pickle.dump(bundle, f)

print("Saved: uav_model_bundle.pkl")

# ============================================================
# 8. EXPORT ONNX (CPU)
# ============================================================
dummy = torch.randn(1, seq_length, 9)

torch.onnx.export(
    model,
    dummy,
    "uav_model.onnx",
    input_names=["input"],
    output_names=["output"],
    dynamic_axes={"input": {0: "batch"}},
    opset_version=12
)

print("Saved: uav_model.onnx")
```

Figure 21 exporting to onnx

```python
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay

# ============================================================
# 1. REGRESSION ACCURACY (TOLERANCE-BASED)
# ============================================================
def regression_accuracy(y_true, y_pred, tolerance):
    return (np.abs(y_true - y_pred) <= tolerance).mean() * 100

tolerance = 0.05  # adjust based on your data scale

overall_acc = regression_accuracy(true_inv, pred_inv, tolerance)
print(f"\nOverall Accuracy (±{tolerance}): {overall_acc:.2f}%")

print("\nPer-output Accuracy:")
for i, col in enumerate(target_cols):
    acc = regression_accuracy(
        true_inv[:, i],
        pred_inv[:, i],
        tolerance
    )
    print(f"{col:>8s} Accuracy: {acc:.2f}%")

# ============================================================
# 2. SEQUENCE-LEVEL ACCURACY (STRICT)
# ============================================================
seq_correct = np.all(
    np.abs(true_inv - pred_inv) <= tolerance,
    axis=1
)
```

Figure 22 regression accuracy

```python
seq_accuracy = seq_correct.mean() * 100
print(f"\nSequence Accuracy: {seq_accuracy:.2f}%")

# ============================================================
# 3. CONFUSION MATRIX (REGRESSION-CORRECT)
# ============================================================
y_true_cls = np.ones(len(seq_correct))     # expected: correct
y_pred_cls = seq_correct.astype(int)       # model outcome

cm = confusion_matrix(y_true_cls, y_pred_cls)

disp = ConfusionMatrixDisplay(
    confusion_matrix=cm,
    display_labels=["Incorrect", "Correct"]
)
disp.plot(cmap="Blues")
plt.title("Regression Confusion Matrix (Tolerance-based)")
plt.grid(False)
plt.show()

# ============================================================
# 4. TRAINING LOSS CURVE
# ============================================================
loss_history = []

for epoch in range(15):
    total = 0
    for xb, yb in train_loader:
        optimizer.zero_grad()
        loss = criterion(model(xb), yb)
        loss.backward()
        optimizer.step()
        total += loss.item()
```

Figure 23 confusion matrix and training loss curve

```python
    avg_loss = total / len(train_loader)
    loss_history.append(avg_loss)
    print(f"Epoch {epoch+1}/15 | Loss {avg_loss:.6f}")

plt.figure(figsize=(6,4))
plt.plot(loss_history)
plt.xlabel("Epoch")
plt.ylabel("MSE Loss")
plt.title("Training Loss Curve")
plt.grid(True)
plt.show()

# ==============================================================
# 5. PREDICTION vs GROUND TRUTH
# ==============================================================
def plot_prediction(idx, name, n=200):
    plt.figure(figsize=(8,4))
    plt.plot(true_inv[:n, idx], label="True")
    plt.plot(pred_inv[:n, idx], label="Pred")
    plt.title(name)
    plt.legend()
    plt.grid(True)
    plt.show()

for i, col in enumerate(target_cols):
    plot_prediction(i, col)
```

Figure 24 prediction vs ground truth

```python
# ==============================================================
# 6. ERROR DISTRIBUTION
# ==============================================================
errors = true_inv - pred_inv

plt.figure(figsize=(6,4))
plt.hist(errors.flatten(), bins=50)
plt.title("Prediction Error Distribution")
plt.xlabel("Error")
plt.ylabel("Count")
plt.grid(True)
plt.show()
```

Figure 25 error distribution

Aditya Roy

Minor Project- 3rd Time

B.Tech- ASE Minor Project

B.Tech.

M. S. Ramaiah University Of Applied Sciences

Document Details

Submission ID

trn:oid:::1:3471114141

Submission Date

Feb 2, 2026, 5:31 PM GMT+5:30

Download Date

Feb 2, 2026, 5:49 PM GMT+5:30

File Name

Aditya_Roy-_minorprojectrewrote_V03-_3rd_Time.docx

File Size

45.0 KB

15 Pages

5,415 Words

30,334 Characters

3% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

17 Not Cited or Quoted 3%
Matches with neither in-text citation nor quotation marks

2 Missing Quotations 1%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

3% Internet sources
2% Publications
0% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Aditya Roy

Minor Project- 3rd Time

B.Tech- ASE Minor Project

B.Tech.

M. S. Ramaiah University Of Applied Sciences

Document Details

Submission ID

trn:oid:::1:3471114141

Submission Date

Feb 2, 2026, 5:31 PM GMT+5:30

Download Date

Feb 2, 2026, 5:50 PM GMT+5:30

File Name

Aditya_Roy-_minorprojectrewrote_V03-_3rd_Time.docx

File Size

45.0 KB

15 Pages

5,415 Words

30,334 Characters

0% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

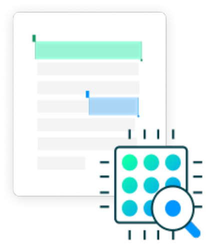
0 AI-generated only 0%
Likely AI-generated text from a large-language model.

0 AI-generated text that was AI-paraphrased 0%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative results), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.



False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.