Proceedings Paper:
Moore, J. and Sawyer, D. (2024) Equipment health monitoring for industrial robotic arms. In: 2024 IEEE 20th International Conference on Automation Science and Engineering (CASE). IEEE International Conference on Automation Science and Engineering (CASE), 28 Aug - 01 Sep 2024, Bari, Italy. Institute of Electrical and Electronics Engineers (IEEE). ISBN: 979-8-3503-5852-0. ISSN: 2161-8070. EISSN: 2161-8089.

https://doi.org/10.1109/CASE59546.2024.10711629

Equipment Health Monitoring for Industrial Robotic Arms*

James Moore & Daniela Sawyer

Abstract— **The topic of equipment health monitoring (EHM) for robotics, including condition monitoring (CM), process monitoring (PM) and predictive maintenance (PdM) is of great interest within the literature, however, there is a significant lack of historical datasets for techniques to be tested upon. Commercial offerings in this area are often manufacturer specific, meaning that fleets that include robots from multiple suppliers cannot easily have performance/condition compared across the fleet. To address this, the work presented within this paper includes an accelerated wear test (AWT) conducted on an industrial robotic arm whilst being monitored using a suite of retrofitted sensors. The resulting data from the AWT is then analysed through a variety of techniques, including regression models, classification models, and a long short-term memory (LSTM) autoencoder, to demonstrate the potential for such methods to be utilised for robot EHM.**

Additionally, the associated dataset captured during the AWT is to be made openly available through the University of Sheffield's online research data repository, ORDA, to allow further research to be conducted.

I. Introduction

It is widely known that robotic arms generally demonstrate good performance when it comes to repeatability, but do not perform as well in absolute accuracy due to the cumulative errors associated with the serial configuration of rotary axes. This has meant they are traditionally employed for lower-precision applications (e.g. welding and manipulation in automotive manufacture). However, due to the advantages in flexibility, larger working volumes, dexterity, and lower costs compared to traditional forms of manufacturing equipment (e.g. CNC milling/turning centres, additive manufacturing systems, etc.), there is a desire from within industry to move to industrial robotic arms as platforms for higher-precision applications [1].

Technologies such as factory-fitted external encoders are gradually becoming a more common option available on robots (e.g. FANUC and MABI) to help improve the absolute accuracies from the order of ±1 mm to ±0.1 mm. This goes some of the way to enabling the adoption of robotic arms for higher-precision operations. However, currently the only widely accepted method for testing the geometric performance of a robot is ISO 9283:1998 [2]. This relies on lengthy and complex measurement routines utilising typically very expensive equipment, such as high-accuracy laser trackers. Having an ongoing measure of a robot's performance is therefore not feasible for most manufacturers, so it is impossible to determine if a robotic arm in is a good enough condition to undertake critical operations until after the fact. By this point any issue with the robot may have been transferred onto the component/assembly which may result in scrap or repair.

There is therefore a need for simpler, more regular performance checks for industrial robotic arms. To this end, the research presented investigates methods of equipment health monitoring (EHM) for robots. The term EHM in this case refers to the use of signals (whether they be sensor-, drive-, or controller-based) to characterise the baseline performance of equipment and provide an indication that its performance may have changed if deviations from this baseline are detected.

This paper is organised as follows: A brief review of related academic literature and commercial solutions are presented in Section II; a description of the equipment used for the trials, and of the experimental trials themselves is provided in Sections III and IV, respectively; Section V discusses the analysis methodology and machine learning techniques tested on the data. The results and discussion are provided in Section VI and concluding remarks in Section VII.

II. Review

A. Academic Literature

There is significant interest within the literature in topics related to EHM for robots, such as CM, PM, predictive maintenance (PdM), and condition-based maintenance (CBM). However, one common theme that was noted is that the lack of historical datasets from industrial robots is a major limiting factor for such research. Although some researchers were able to source historical industrial robot datasets, while others were able to develop techniques by employing either datasets captured from laboratory-based robotic arms, or by focussing on specific components from robotic arms.

For the most part, the literature favours a combination of vibration monitoring of gears (resolvers) and/or motors through the use of accelerometers (e.g. Nentwich and Reinhart [3], Uhlmann et al.[4], and Wang et al.[5]); as well as internal PLC/controller signal monitoring, such as joint motor torque or motor current (e.g. Izagirre et al.[6], Panicucci et al.[7], and Yang et al.[8]). This outcome is perhaps unsurprising, as similar techniques are widely employed for diagnostics for other industrial equipment and machinery (pumps, turbines, machine tools, etc).

There is a wide range of models presented within the literature, however, they generally fall into two categories:

*Research conducted as part of the AIRLIFT (Additive IndustRLIsation Future Technology) Programme, funded by the Aerospace Technology Institute.

J. Moore and D. Sawyer are both with the University of Sheffield Advanced Manufacturing Research Centre's Integrated Manufacturing Group, Factory 2050, Europa Way, Sheffield, S9 1ZA, UK (phone: 0114 215 8200; e-mail: j.moore@amrc.co.uk, d.sawyer@amrc.co.uk).

physics-based, and data-driven. Although Aivaliotis et al.[9], [10] were able to develop a sophisticated physics-based model that included components such as bearing degradation curves, the work the authors presented highlights the complexity of developing such models. It is likely that adapting such models for various configurations of robot would be extremely involved and time-consuming. It is for such reasons that Chen et al.[11] state the literature tends to favour data-driven models.

The data-driven models identified were comprised of statistical models and machine learning (ML) models. Statistical models (e.g. statistical process control (SPC)) are widely used in industry and offer a relatively simple method for monitoring and controlling a process. This type of model has been demonstrated to be successful in detecting faults in robots by Jaber & Bicker [12] and Yang et al.[8]. It should be noted that statistical models are heavily reliant on selecting the most appropriate signal feature(s) to be successful in detecting the targeted fault(s).

The strength of ML-based models is the ability to automatically identify relationships within datasets and therefore extract the most relevant indicators/features that correspond to known faults (if datasets are suitably labelled). However, such techniques are often data-hungry, and require relatively large datasets to accurately characterise these relationships. As mentioned above, such datasets are few and far between. Nonetheless, techniques such as data augmentation and transfer learning can be employed to help alleviate this shortcoming.

Previous work conducted by the University of Sheffield Advanced Manufacturing Research Centre (AMRC) has already demonstrated the capability of ML for EHM and PM of machine tools [13]. Various ML techniques were tested including traditional ML models, as well as convolutional neural networks (CNNs) – a form of deep learning; all of which were successful in providing accurate classifications of various failure modes of the machine tool and workpiece.

Ideally, the data and models mentioned above could be used to develop prognostic tools, such as remaining useful life (RUL). However, this generally requires in-depth knowledge about component degradation, or extensive run-to-failure data, which again, is limited for industrial robots. Although a few examples of these kind of tools being developed for robots were demonstrated in the literature, these often could not be validated on real-world failures. Only one exception to this, provided by Yang et al.[8], was identified where the authors were able to develop a domain-generalisation adversarial long short-term memory (DGALSTM) model that provided RUL estimations on data captured from display screen transfer robots that included real-world servo motor failures.

B. *Commercial Solutions*

In terms of commercially available solutions for robotic EHM, there is little that could be identified beyond the products developed by the robot manufacturers. ABB have two tools that fall within this category, Service Information System (SIS) and Mechanical Condition Change (MCC). SIS can provide information to help highlight which robots might be under most stress and help inform which may require more regular maintenance. However, this system is based purely on the historical log of the amount and severity of work a robot has undertaken (i.e. "duty factor"), rather than utilising any monitoring methods to provide live health indicators [14]. MCC on the other hand helps identify a change in a robot's condition through a short routine that moves each axis in turn. Torque energy and torque noise (equivalent to total torque and torque signal variability, respectively) can be compared each time the routine is run with the aim of detecting any issues or change in the motor or gearbox [15]. It should be noted that this approach is reminiscent of the fingerprint routine approach developed for machine tools by the AMRC [13].

KUKA provide a fleet monitoring platform for their robots called iiQoT. It is claimed that tools such as condition monitoring and fault diagnosis are included within this system [16], but no details on the techniques used, or the output of the system are provided.

Despite the solutions from ABB and KUKA being available for their respective robotic arms, these are not solutions that could be used across a fleet made up of robots from a range of manufacturers – meaning that it would make it near-impossible to directly gauge health/performance across the entire fleet. For this, a more generic or third-party solution would be required. This may become a more pertinent issue in the coming years as the availability and use of low-cost robots become more ubiquitous throughout manufacturing.

III. Equipment

This sections below describe the equipment used for the trials, including details on the industrial robotic arm (Section A), and sensors and data acquisition hardware (Section B).

A. *Industrial Robotic Arm*

The robot selected for testing was a KUKA KR16 L6 with KRC2 controller, which, despite being in full working order, had been designated as being at the end of its useful life within the AMRC. This was advantageous as it was planned that the robot would undergo accelerated wear testing (AWT) which would ideally lead to some form of degradation which would not subsequently require repairing upon completion of the trials.

To simplify the testing and analysis required during this project, it was decided that only two of the robot's six joints should be monitored: joint 3 as the joint under test, and the joint 2 as a control. These joints were chosen as they have the same configuration as each other, as well as being representative of joint configurations on other robotic platforms.

To maximise the stress placed on the robot during testing, payloads were fabricated to achieve the maximum payloads for the end effector and supplementary payload mounting positions (6 kg and 10 kg limits, respectively). The end effector payload was machined from an aluminium block, whilst the supplementary payload was cut from box section steel; each with mounting holes drilled to allow the necessary bolts to be fitted. These are illustrated in Fig. 1.

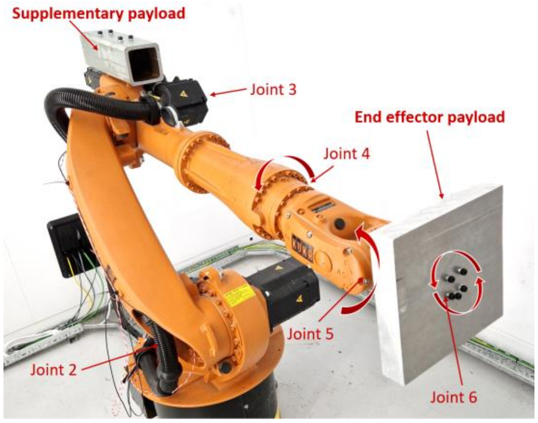

Figure 1. The KUKA KR16 L6 robot with payloads and axis naming.

B. Sensors and data acquisition hardware

As discussed in Section II, a combination of tip- and joint-mounted accelerometers, motor power monitoring and joint resolver monitoring were identified as providing the best detection capability for EHM of a robotic arm. Due to the age of the KUKA, it was not feasible to collect high frequency data directly from its KRC2 control unit, meaning resolver data could not be captured, nor internal motor power. However, it was possible to conduct current monitoring by installing a current transducer (current clamp) around one of the three phase cables running between the motor drive unit, and the joint motor itself. For this task, a LEMO AT 10 B10 current transformer was installed on one of the phase cables for both joint 2 and 3 directly below each of the motor drive units within the control cabinet. An additional current clamp was also added to joint 6 for the triggering purposes.

To capture motor and gearbox vibration of both the joints, a tri-axial accelerometer was adhesively mounted to the link arm mounting plate for each joint. The accelerometers used were MMF KS903B.100, that have a frequency range of 0.15-12,000 Hz and sensitivity of 100 mV/g. To help ascertain if ambient temperature or the temperature of each joint has a notable effect on the robot's characteristics, self-adhesive resistance temperature detectors (RTDs) were also mounted to both joints, as well as to the robot cell wall. An example of the mounting of these sensors is provided in Fig. 2.

The signals from the sensors were captured by a National Instruments (NI) cDAQ-9174 loaded with two NI 9234 sound and vibration input modules for the six accelerometer channels; one NI 9217 RTD input module for the three RTD channels; and one NI 9201 voltage input module for the three current clamp channels. Data was logged on a local PC running NI's FlexLogger software with the sampling rates as shown in Table I.

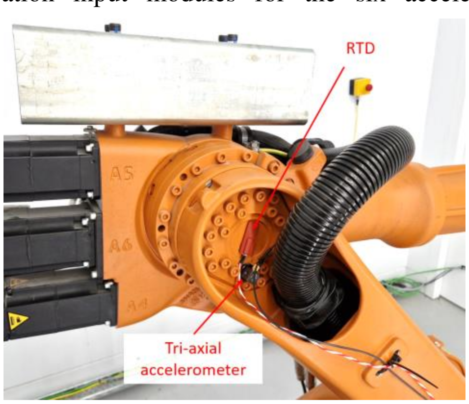

Figure 2. Sensor mounting on joint 3.

IV. METHOD

The following sections describe the robot's routine for the accelerated wear trials (Section A), and as no catastrophic failure was induced, the additional interventions undertaken to attempt to accelerate the degradation of the robotic arm further (Section B).

A. Accelerated Wear Test

The robot underwent AWT, whereby joint 3 was rotated in isolation back and forth across 284° of its travel (between -130° and 154°) at 100% speed and acceleration continuously (i.e. 24 hours a day, seven days a week). The AWT was run for a total of 28 weeks. There were only brief intermittent breaks every 1,000 repetitions of this motion, in which a fingerprint routine could be run to gauge any change in performance.

The fingerprint routine itself consisted of three repetitions of the same motion conducted on joint 3, repeated at 10%, 50% and 100% speed. A similar motion was then conducted on the control joint, joint 2, (travelling between -153° and -46°) at the same speed intervals.

Due to the extended period over which this testing was conducted, data could only feasibly be captured during these fingerprint routines. It was also imperative that the whole process could be left to run unmanned, including the data capture, so a method for automatically triggering the data capture was required. To do this, a short rotation of joint 6 was programmed at the very start of the fingerprint routine. It was then possible to set up a trigger condition within the FlexLogger software that could initiate the data capture based on the rising edge of the joint 6 current clamp signal. As the fingerprint routine was repeatable in duration, the acquisition window was fixed at 80 s, matching the length of the routine.

B. Additional Interventions

Once 12 weeks had passed, no noticeable change in the performance of the robot had been identified (i.e. no significant change in the sound or operating temperatures; and nothing visual was apparent). It was therefore decided that it would be pertinent to implement intermittent cool-downs at 1-2 week intervals, as well as additional steps towards the end of the 26 week period that could be employed to accelerate any degradation further.

The cool-down procedure meant stopping the robot and allowing the robot to cool down to ambient temperature, before resuming the accelerated wear routine. The reasoning

TABLE I. SAMPLING RATES FOR DATA ACQUISITION

NI Module	***Sampling rate (Hz)***
NI 9234 (Accelerometers)	25,600 (10,240[a])
NI 9217 (RTDs)	1
NI 9201 (Current clamps)	1,000

a. Lower sampling rate used for the first 228 instances to test robustness of system.

behind this was that although running the robot continuously was observed to be generating significant amounts of heat in joint 3, up until that point joint 3 had been running at a relatively stable temperature (between 60°C-70°C as measured by the RTD). The cool-downs were therefore intended to present an extra stress by thermally cycling the robot.

Shortly after the second cool-down procedure had taken place (around 14 weeks into the AWT), oil was noticed to be dripping onto the floor from joint 3. As this was the type of natural degradation that could potentially occur within a factory without being noticed, the robot was left to continue the AWT without any maintenance (i.e. repair of any seals, or refilling the oil) in the hope that this could instigate further, more serious degradation.

Towards the end of the AWT period (~26 weeks), there had still been no further obvious degradation, or catastrophic failure to the robot – although oil did continue to leak slowly for several weeks after it was first noticed. It was therefore decided that further intervention should be conducted upon the robot to accelerate any further degradation. This was done in two ways: draining the remaining oil from joint 3; and conducting a controlled crash.

1) Draining Oil

At week 27, all the remaining oil (108 ml) was drained from the joint 3 gear unit. To then accelerate the effects of the oil leak, only half of this oil (55 ml) was returned to the gear unit before it resumed the AWT routine. After another three days of testing, no significant change in the robot's performance was observed, so the remaining oil was drained, and the gear unit left empty upon returning to the AWT routine for another two days of testing.

2) Controlled Crash

It was decided that conducting a controlled crash of the robot would be a suitable final test that could be representative of a real-world event in industry that could affect a robot's performance. The crash was intended to focus as much of the force of the impact on joint 3. To do this, the end effector payload was removed and joint 5 was rotated to be perpendicular to the arm to ensure the force of the impact was not absorbed in damaging the payload fixing bolts or snapping the drive belt. To provide a relatively rigid, immoveable target, a cast iron machining tombstone was moved into the cell with a steel plate laid on top.

As additional fingerprint routines were required after the crash had occurred to allow data to be captured on the robot's subsequent performance, the crash needed to be not too severe that it would incapacitate the robot. Equally, it needed to be severe enough to make it likely that some mechanical damage would be sustained. It was therefore decided that the controlled crash would involve a series of crashes, with each collision incrementing in speed from 5% to 25% of joint 3's maximum rotational speed; equating to 107 mm/s to 535 mm/s linear speed at the end effector.

After the final crash, there was a subtle grinding and clicking noise present when moving joint 3 that could not be heard before. This was deemed a suitable point to conclude the crash testing, as it was thought that this could be an indication of damage to the gear unit. The tombstone was removed from the cell, the end effector payload reinstalled, and the AWT routine was resumed for a further six days to gather the post-crash fingerprint data.

V. Analysis

This section presents the methodology conducted in the data preparation (Section A), data cleansing (Section B), data labelling (Section C), and training of ML models (section D).

A. Data preparation

Each of the 3621 recordings contained all the signals recorded during a single fingerprint routine. As described in Section IV, this comprised of a motion on each joint 3 and joint 2 in isolation, run at 10%, 50% and 100% speed – giving six individual motions. Rather than calculating a single set of features on each fingerprint, the signals were segmented into the six motions, to allow a set of features to be extracted from each. This is illustrated for the accelerometer data in Fig. 3.

The features calculated in this project were those recommended by Saidi et al. [17] for the diagnosis of wind turbine bearings. The equations to calculate these features are given in Equations 1 to 11:

Mean $$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \quad (1)$$

Standard deviation $$\sigma = \left(\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})\right)^{\frac{1}{2}} \quad (2)$$

RMS $$\left(\frac{1}{N}\sum_{i=1}^{N} x_i^2\right)^{\frac{1}{2}} \quad (3)$$

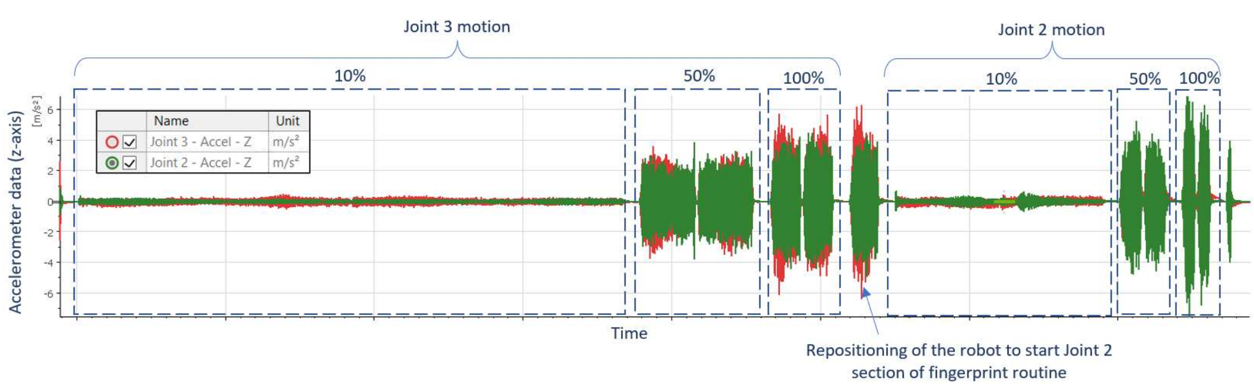

Figure 3. Example of the segmentation of accelerometer data for a single fingerprint routine with each motion labelled.

Kurtosis	$\frac{1}{N}\sum_{i=1}^{N}\frac{(x_i-\bar{x})^4}{\sigma^4}$	(4)
Skewness	$\frac{1}{N}\sum_{i=1}^{N}\frac{(x_i-\bar{x})^3}{\sigma^3}$	(5)
Peak to peak	$x_{max}-x_{min}$	(6)
Crest factor	$\frac{x_{max}}{RMS}$	(7)
Shape factor	$\frac{\mathrm{RMS}}{\frac{1}{N}\sum_{i=1}^{N}\lvert x_i\rvert}$	(8)
Impulse factor	$\frac{x_{max}}{\frac{1}{N}\sum_{i=1}^{N}\lvert x_i\rvert}$	(9)
Margin factor	$\frac{x_{max}}{\left(\frac{1}{N}\sum_{i=1}^{N}\lvert x_i\rvert\right)^2}$	(10)
Energy	$\frac{1}{N}\sum_{i=1}^{N}{x_i}^2$	(11)

These statistical features were calculated in the time-domain for each signal (except RTD channels which only included maximum, minimum, mean, and standard deviation), in each motion – yielding a total of 696 features for each fingerprint. These were calculated for all fingerprint routines and consolidated into a feature table, with each feature having its own column, and each row representing each fingerprint run. This format also made it possible to easily produce comparison plots of the various features, as well as utilise MATLAB's ML toolboxes (i.e. Regression Learner and Classification Learner) which require such a format.

B. Data Cleansing

After an initial exploration of the data, it became apparent that there were some entries that appeared anomalous/spurious and so some cleansing of the data was necessary. A summary of these actions is presented below:

- Several features showed sensitivity to operating temperature, so to remove the effect of cool-downs, any fingerprint routines that had taken place where the mean joint 3 temperature was below 39.5°C were excluded from the analysis.
- A significant step-change in the output of the joint 2 accelerometer at around 4200 hours of the AWT was observed. This was not related to any known event occurring on the robot, and no change in performance was observed over this time. All features calculated from this accelerometer were therefore omitted from the analysis.

C. Data Labelling

For simplifying manual analysis, as well as training ML models (particularly those produced using supervised ML techniques) the feature tables would require labels of the ground truth. The most straightforward approach was to include an 'Elapsed time' column which was calculated from the difference in hours between the timestamp of each fingerprint routine and that of the first fingerprint run. This would be useful in creating plots, as well as in training regression models, which generally require a continuous output/response.

Additionally, a column containing categorical labels was also included for the purpose of training classification type ML models. These were based on the various stages and observations from the AWT:

None – the first 1456 fingerprint routines where no significant change or event had been observed.

Oil leaking – the 1598 fingerprint routines where oil had been observed 'naturally' leaking from the robot.

Oil partially drained – the 57 fingerprint routines conducted after half of the remaining oil had been drained from the gear unit.

No oil – the 34 fingerprint routines conducted after the oil had been completely drained from the gear unit.

After crash – the final 226 fingerprint routines conducted following the controlled crash.

D. Machine Learning

The approach taken within this phase of work was exploratory, and a wide range of techniques were employed to provide indications of certain areas that could be promising for further development in robot EHM. It should be noted that due to this style of approach, combined with a limited time frame, it was not possible to be exhaustive in the testing of the various techniques.

1) Regression

This type of model is useful for EHM when trying to make predictions such as time to failure or RUL. MATLAB's 'Statistics and Machine Learning Toolbox' includes the 'Regression Learner' application, which automates the process of training a range of regression model types, including linear regression, regression trees, support vector machines (SVMs), Gaussian process regression, kernel approximation regression, and neural networks.

Although the robot did not suffer a catastrophic failure during the trials, if regression models could be used to correctly predict the elapsed time, this indicates that gradual changes in the signals that might be associated with wear could be used to predict failures in the future. As the potential sudden changes caused by the draining of oil, or crash testing, are not representative of natural degradation (and hence RUL would not be appropriate for detecting such events), these final stages were omitted and only the data from the first 3054 fingerprint routines were used for this analysis. Cross-validation with five folds was utilised to prevent overfitting, and 10% of the data was also held back as an unseen test data set for once the training process had been completed.

To help reduce the computational effort in training the models, overfitting, as well as noise from features that have little relationship to the condition of the robot, it was possible to utilise the application's feature selection tool. This utilises the minimum redundancy maximum relevance (MRMR) algorithm for feature ranking to assess the importance of each feature, as well as reduce redundancy within the feature set.

2) Classification

Classification models are used to predict a discrete classifier (or label) from a set of input data. This could be useful in EHM for providing detection of known faults. MATLAB's 'Statistics and Machine Learning Toolbox' also includes the 'Classification Learner' application, which automates the process of training a range of regression model types, including decision trees, discriminant analysis, support vector machines, logistic regression, nearest neighbours, naive Bayes, kernel approximation, ensemble, and neural networks.

This method of analysis was conducted to test if the data would be suitable for identifying various conditions of the robot throughout the AWT, and if so, which type of model provides the most success in doing so.

Like the regression analysis, reducing the number of features was also trialled. This was done using the feature selection tool within the 'Classification Learner' application, with the MRMR algorithm. And again, cross-validation with five folds was utilised, and 10% of the data was held back as an unseen test data set for once the training process had been completed.

3) Long Short-Term Memory (LSTM) Autoencoder

An LSTM autoencoder is a hybrid model of an LSTM neural network and autoencoder, which is also based on a neural network. The autoencoder learns to encode input data into a lower dimensional (compressed) representation, and then decode this back to reconstruct the original higher dimensional input. The LSTM portion of the model makes the autoencoder capable of dealing with a sequence of data, such as time series. This means that rather than processing a data point in isolation, the 'memory' allows it to consider how this relates to the previous data point, which is often crucial in time-series data. By measuring the error in the model's reconstruction, it is possible to detect changes or anomalies in the input data, as is demonstrated by Jeon et al for anomaly detection in the operation of quadcopters [18].

One of the advantages of LSTM autoencoders is that they are semi-supervised, meaning they are only trained on examples of 'good' data, and therefore do not require extensive examples of 'bad' data – which is often what is lacking when it comes to data sets from industrial equipment, such as robots. The process followed for training and testing the LSTM autoencoder was based on that set out in the MathWorks example "Anomaly Detection in Industrial Machinery Using Three-Axis Vibration Data" [19], including the definition of the LSTM autoencoder structure, which is shown in Table II.

TABLE II. LSTM AUTOENCODER STRUCTURE

Layer	Size
Sequence input layer	1-dimension
Bi-LSTM layer	16 hidden nodes
ReLU layer	-
Bi-LSTM layer	32 hidden nodes
ReLU layer	-
Bi-LSTM layer	16 hidden nodes
ReLU layer	-
Regression layer	**-**

The LSTM autoencoder was trained on the data that had the label 'None', i.e. the data that was captured with no fault on the robot observed. Validation was not used within the training process, but 10% of the data was excluded from the training data for validation purposes. Once trained, the model could then make predictions on the validation data, and the reconstruction error could be found by calculating the root mean squared error (RMSE) between the input data and the reconstructed data. By observing the difference in the reconstruction errors produced by the 'None' data and for the remaining labels, a threshold could be manually set at which an anomaly would be classified.

VI. RESULTS AND DISCUSSION

Due to the range of results and plots produced in the exploratory analysis of this data, it is not practicable to present a comprehensive record of this within this paper. Instead, an overview of the findings that are most pertinent to EHM from regression, classification, and LSTM autoencoder are presented in Sections A, B and C, respectively.

A. Regression

It can be seen in Fig. 5 that several regression models provide accurate predictions of the elapsed time, with Bagged Tree ensemble achieving the lowest test data RMSE value of 30.74 hours (i.e. less than 1% error with respect to the duration of the AWT). No catastrophic failure resulted from the AWT, so it was not possible to set up a RUL or time to failure prediction model. However, as these regression techniques have proved capable in predicting the elapsed time with good accuracy, it is conceivable that when being used on a data set that includes component failure that they could be adapted relatively easily to provide RUL predictions.

B. Classification

The classification models tested resulted in very good performances, with most approaching or achieving 100% test data accuracy, with only a few exceptions (see Fig. 6).

C. LSTM Autoencoder

The LSTM autoencoder approach was tested using nine features all from the joint 3 motion selected through a trial-and-error approach – listed in Table III. The anomaly detection threshold was also selected through trial and error, with 0.4 (reconstruction error) found to be the most accurate.

TABLE III. LSTM AUTOENCODER FEATURES

No.	*Feature*
1	Energy of joint 3 z-axis accelerometer at 100% speed
2	Shape Factor of joint 3 z-axis accelerometer at 100% speed
3	Kurtosis of joint 3 z-axis accelerometer at 100% speed
4	Kurtosis of joint 3 z-axis accelerometer at 10% speed
5	RMS of joint 3 z-axis accelerometer at 10% speed
6	RMS of joint 3 y-axis accelerometer 50% speed
7	Margin factor of joint 3 z-axis accelerometer at 50% speed
8	RMS of joint 3 z-axis accelerometer at 50% speed
9	RMS of joint 2 current draw at 10% speed

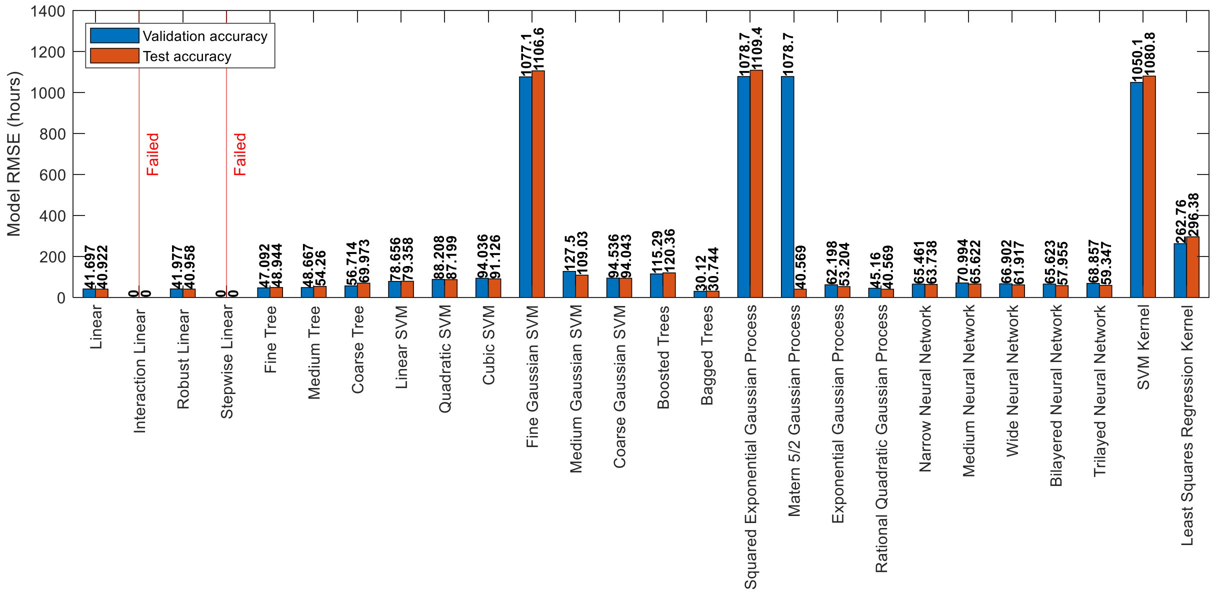

Figure 5. Regression model RMSEs (lower values indicate more successful models).

The resulting model provided a relatively accurate (98.52%) method for detecting anomalies on the robot from only a small feature set – illustrated in Fig. 7. Additionally, this technique has the major advantage of only being trained on 'normal' data, meaning there is no requirement for examples of abnormal/faulty data to be provided during the training stage.

Obviously, it is beneficial to have some examples of fault data to be able to test the accuracy of the model upon, however, these do not have to be in large numbers as is normally necessitated by other ML techniques. As it has already been highlighted in Section II, there is a significant lack of data from industrial robots, hence techniques that require little or no abnormal/faulty data examples such as LSTM autoencoders are desirable.

VII. Conclusions

This paper has presented a range of techniques that have been tested for the purpose of EHM in industrial robotic arms, including regression models, classification models, and an LSTM autoencoder. It has been found that regression models were accurate in predicting the elapsed time of the AWT with RMSE values down to 30.74 hours in the case of Bagged Tree Ensemble. These models have the potential to be adapted for RUL predictions on other similar data sets which do include examples of catastrophic failures.

Most of the classification models tested achieved close to 100% accuracy in classifying the condition of the robot. However, the biggest weakness of these techniques is that they ultimately require a suitable number of examples of 'faulty' data and would likely not perform well if any unseen fault occurs.

To try and mitigate for this limitation in classification techniques, a LSTM autoencoder model was also tested. This only requires examples of 'normal' data for training, and so is a potentially desirable technique for use on industrial robots where there is a notable lack of historical data sets. With a small feature set, consisting of only nine features selected through trial-and-error, it was possible for the LSTM autoencoder to achieve 98.52% accuracy in detecting anomalous data (i.e. data captured after the oil leak occurred). It might be possible to increase the accuracy of this further

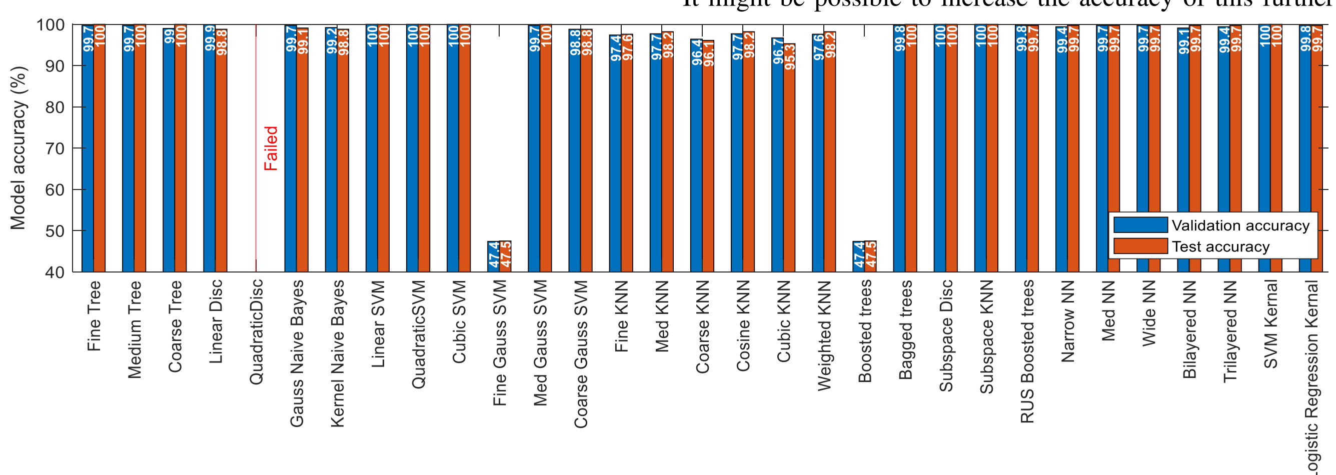

Figure 6. Classification model accuracies (higher values indicate more successful models).

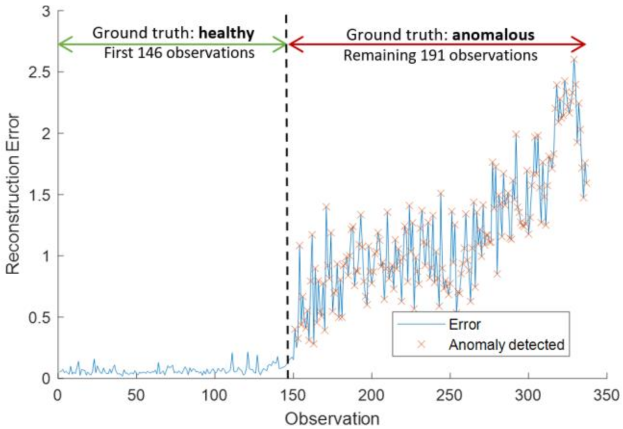

Figure 7. Illustration of LSTM autoencoder performance on test data.

with additional feature selection and model structure optimisation.

Below are areas that have been identified for further investigation based on the research presented in this report:

- Develop a system that can provide an online, real-time output based on the fingerprint routine.
- Investigate how data streams from more modern robot controllers could be included within such a system.
- Test the developed techniques on data sets that include catastrophic component failure to develop RUL capability.
- Validate the approach on a robot employed within industry.

By continuing development in the above areas, it may be possible for the research presented to be eventually translated into an industrial solution. Such a solution could be employed on robots that require a high confidence of their performance/condition immediately prior to conducting critical, high-precision operations, thus reducing the risk to the workpiece.

It could also be used regularly on a fleet of robots to help gauge how the performance of each robot progresses over time. Significant or accelerated changes could provide early detection of impending failures, allowing maintenance to be scheduled in a timely manner (i.e. CBM).

If it were possible to also develop RUL capability as part of this industrial solution, this would enable scheduling of maintenance activities well in advance based on each robot's actual condition (i.e. PdM). This would reduce unplanned downtime due to early component failure, as well as reduce unnecessary planned downtime due to premature preventative maintenance activities.

Finally, the dataset produced during the AWT is to be made available through the University of Sheffield's online research data repository, ORDA. It is hoped this dataset will go some way in helping to fill the gap in openly available performance data for industrial robots; providing a platform for other researchers to analyse, develop and test their own models upon.

Acknowledgments

The authors would like to thank GKN Aerospace in supporting this research as part of the AIRLIFT (Additive IndustRLIsation Future Technology) programme. For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising from this submission.

References

[1] C. Cristalli, L. Lattanzi, D. Massa, and G. Angione, "Cognitive Robot Referencing System for High Accuracy Manufacturing Task," *Procedia Manufacturing*, vol. 11. pp. 405–412, 2017.

[2] British Standards Institution, "BS EN ISO 9283:1998 - Manipulating industrial robots - Performance criteria and related test methods." 1998.

[3] C. Nentwich and G. Reinhart, "A Method for Health Indicator Evaluation for Condition Monitoring of Industrial Robot Gears," *Robotics*, vol. 10, no. 2, p. 80, 2021.

[4] E. Uhlmann, J. Polte, and C. Geisert, "Condition Monitoring Concept for Industrial Robots."

[5] J. Wang, D. Wang, and X. Wang, "Fault Diagnosis of Bearings Based on Multi-Sensor Information Fusion and 1D Convolutional Neural Network," in *Proceedings of the 39th Chinese Control Conference*, 2020, pp. 3087–3091.

[6] U. Izagirre, I. Andonegui, A. Egea, and U. Zurutuza, "A methodology and experimental implementation for industrial robot health assessment via torque signature analysis," *Appl. Sci.*, vol. 10, no. 21, pp. 1–14, 2020.

[7] S. Panicucci *et al.*, "A cloud-to-edge approach to support predictive analytics in robotics industry," *Electron.*, vol. 9, no. 3, pp. 1–22, 2020.

[8] Q. Yang *et al.*, "Fault prognosis of industrial robots in dynamic working regimes: Find degradation in variations," *Meas. J. Int. Meas. Confed.*, vol. 173, no. October 2020, p. 108545, 2021.

[9] P. Aivaliotis, Z. Arkouli, K. Georgoulias, and S. Makris, "Degradation curves integration in physics-based models: Towards the predictive maintenance of industrial robots," *Robot. Comput. Integr. Manuf.*, vol. 71, p. 102177, 2021.

[10] P. Aivaliotis, K. Georgoulias, and G. Chryssolouris, "The use of Digital Twin for predictive maintenance in manufacturing," *Int. J. Comput. Integr. Manuf.*, vol. 32, no. 11, pp. 1067–1080, 2019.

[11] T. Chen, X. Liu, B. Xia, W. Wang, and Y. Lai, "Unsupervised anomaly detection of industrial robots using sliding-window convolutional variational autoencoder," *IEEE Access*, vol. 8, pp. 47072–47081, 2020.

[12] A. A. Jaber and R. Bicker, "Wireless fault detection system for an industrial robot based on statistical control chart," *Int. J. Electr. Comput. Eng.*, vol. 7, no. 6, pp. 3421–3435, Dec. 2017.

[13] J. Moore, J. Stammers, and J. Dominguez-Caballero, "The application of machine learning to sensor signals for machine tool and process health assessment," *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.*, p. 095440542096089, Sep. 2020.

[14] ABB AB Robotics, "Condition Based Maintenance through SIS Analysis." ABB AB Robotics, 2021.

[15] ABB AB Robotics, "Application manual - Mechanical Condition Change (MCC)," no. L. ABB AB Robotics, VÄSTERÅS, Sweden, 2018.

[16] KUKA Robotics UK Ltd, "KUKA iiQoT: data-based added value through IIoT for your robots | KUKA AG." [Online]. Available: https://www.kuka.com/en-gb/products/robotics-systems/software/cloud-software/iiqot. [Accessed: 27-Sep-2021].

[17] L. Saidi, J. Ben Ali, E. Bechhoefer, and M. Benbouzid, "Wind turbine high-speed shaft bearings health prognosis through a spectral Kurtosis-derived indices and SVR," *Appl. Acoust.*, 2017.

[18] S. Jeon, J. Kang, J. Kim, and H. Cha, "Detecting structural anomalies of quadcopter UAVs based on LSTM autoencoder," *Pervasive Mob. Comput.*, vol. 88, p. 101736, Jan. 2023.

[19] MathWorks Inc., "Anomaly Detection in Industrial Machinery Using Three-Axis Vibration Data." [Online]. Available: https://uk.mathworks.com/help/predmaint/ug/anomaly-detection-using-3-axis-vibration-data.html. [Accessed: 03-Jul-2023].