Interpretable Fault Detection Approach With Deep Neural Networks to Industrial Applications

Fatemeh Kakavandi[1*], Peihua Han[2*], Roger de Reus[3], Peter Gorm Larsen[1], Houxiang Zhang[2]
[1]*Department of Electrical and Computer Engineering, Aarhus University*, Aarhus, Denmark
[2]*Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology*, Ålesund, Norway
[3]*Device Manufacturing Development Department 2540, Novo Nordisk A/S*, Hillerød, Denmark
Email: fateme.kakavandi@ece.au.dk, peihua.han@ntnu.no

***Abstract*—Different explainable techniques have been introduced to overcome the challenges in complex machine learning models, such as uncertainty and lack of interpretability in sensitive processes. This paper presents an interpretable deep-leaning-based fault detection approach for two separate but relatively sensitive use cases. The first use case includes a vessel engine that aims to replicate a real-life ferry crossing. Furthermore, the second use case is an industrial, medical device assembly line that mounts and engages different product components. In this approach, first, we investigate two deep-learning models that can classify the samples as normal and abnormal. Then different explainable algorithms are studied to explain the prediction outcome for both models. Furthermore, the quantitative and qualitative evaluations of these methods are also carried on. Ultimately the deep learning model with the best-performing explainable algorithm is chosen as the final interpretable fault detector. However, depending on the use case, diverse classifiers and explainable techniques should be selected. For example, for the fault detection of the medical device assembly, the DeepLiftShap algorithm is most aligned with the expert knowledge and therefore has higher qualitative results. On the other hand, the Occlusion algorithm has lower sensitivity, and therefore, higher quantitative results. Consequently, choosing the final explainable algorithm compromises the qualitative and quantitative performance of the method.**

***Index Terms*—Fault detection, Deep neural network, Explainable artificial intelligence, Qualitative and quantitative evaluation, Infidelity, Sensitivity.**

I. Introduction

Evaluating the quality of a process or product and avoiding abnormal chaotic situations are essential tasks in various industries that different conventional and artificial intelligence-based solutions have addressed [1]. Deep learning-based fault detection methods have thrived with the development of sensory systems and the availability of large volumes of data [2]. However, these complicated black box models are challenging to interpret and explain in industrial use cases. For example, in Pharmaceutical manufacturing processes, the quality of the products is critical, and using black box models without any interpretability is challenging [3]. Comparably, controlling a vessel's movement is a delicate task with high sensitivity, and we should ensure that the engine is in its normal status continuously.

Explainable Artificial Intelligence (XAI) algorithms are proposed to add interpretability to Deep Neural Networks (DNN) to gain trust and validity [4]. Furthermore, these methods highlight the contributors to output prediction in complicated DNN models [5]. The highlighting is done by attributing values to input features in DNN, which indicate the feature's importance. However, the interpreting methods are black box themselves, mainly applied in the image processing domain, and rely on the visual interpretability of the image input [6]. Moreover, little work related to the time series has been carried out [7], [8]. Therefore, these methods' performance for time-series data should also be evaluated and reported.

Qualitative and quantitative methods are considered to evaluate the performance of XAI attribution methods. Qualitative methods focus on how accurate the attribution distribution is, which means how the attribution aligns with the expectations. However, for quantitative evaluation methods, the focus is on the performance of attribution methods regarding sensitivity and fidelity [9]. The challenge in XAI methods for time series is that the qualitative evaluation needs some preliminary knowledge about the importance of different input features.

The contribution of this paper is to develop two fault detection models for two different yet similar in nature use cases, a vessel engine and a pharmaceutical assembly process, and employ the best-performing attribution method to explain the prediction function. Furthermore, two DNN models, a Fully Convolutional Network (FCN) and Long Short-Term Memory (LSTM), and different XAI methods have been investigated to achieve a mature explainable fault detection model. Ultimately, we compare the performance of different DNN models and XAI methods and choose the most accurate DNN and XAI methods. The results show that this methodology is applicable to detect fault for diverse cases where multiple sensors are recorded; however, depending on each use case, various DNN models and XAI methods would perform better.

The remainder of the paper is structured as follows. First, background on XAI methods is provided in Section II. Then, the adopted methodological steps in detecting the fault and interpreting the decision-making are described in Section III. Furthermore, the method evaluation is provided in Section section IV, where the industrial use cases are introduced, and the related results are provided and compared. Ultimately, we

* These authors contributed equally to the paper

conclude the final results and findings in Section V.

II. Background

Deep learning-based fault detection can be viewed as time series classification, e.g., binary or multi-class classification using time series data based on the number of categorical faults. This section will briefly introduce the background of XAI methods for time series, the existing challenges, different types of explanation algorithms, and the evaluation methods.

A. XAI on time series classification

XAI research on deep learning models has mainly focused on computer vision and natural language processing. Most existing XAI methods highlight the parts of the input responsible for prediction [10], [11]. This might be easy to understand for images or languages but not for time series data due to the unintuitive nature of time series. Nonetheless, these methods are directly applicable to time series data, e.g., Wang et al. [12] use the class activation map [13] to highlight the regions in the input univariate time series that have the most significant impact on output classification prediction. Moreover, Kashiparekh et al. [14] use the simple perturbation method that occludes parts of the time series and computes the difference in the probability for the predicted class. Regardless of the XAI method used, these methods ultimately create an attribution map that reflects the relevance of input to output. For time series data, this attribution map is often combined with a line plot of the original time series. Experts with domain knowledge are necessary to inspect and verify the explanations.

B. Local and global explanations

The explanations are usually divided into local and global explanations. The explanations are qualified as local when they are valid for a specific sample and global when they are valid for a set of samples or the entire dataset [7]. From this perspective, most XAI methods for deep learning model only provides local explanations. Nonetheless, the local explanations can be used to generate global explanations, e.g., averaging the local explanations to all samples in the dataset [15]. From our experience in fault detection, when it comes to global explanations, it is easier to evaluate the correctness of explanations since the features responsible for specific faults can be identified through domain knowledge. In addition, the model's trustworthiness increases if the global explanations match the expert's knowledge. The local explanation may be more interesting after deploying the model since it can help engineers understand why faults occurred.

C. Evaluating time series explanations

Empirical evaluation of XAI methods is problematic since it is challenging to distinguish errors in the model from errors of the attribution method explaining the model [11]. For this reason, the used model is often included in the evaluation process. The predominant evaluation of explanations has been qualitative evaluation [14], [15]. It is a subjective measure that domain expert assesses the explanations. However, quantitatively evaluating the explanations is objective, critical and challenging. Therefore one way is to verify whether the explanation mechanism satisfies certain axioms such as completeness, fidelity, sensitivity and etc. [9], [16].

III. Methodology

This section provides the methodology we used for detecting the fault and explaining the reason behind the decision, as shown in Fig. 1. First, the DNN model (like FCN) is trained on the data set and predicts the label for each sample as output. Then the XAI method (Gradient) calculates global and local relevance values to different input features explaining the support for decision-making. Furthermore, the attribution algorithm uses the weights, biases, input sample, and output prediction of the DNN model to achieve the results. In addition, various visualization tools are employed to show the global explanation of different features and local relevance for data points in a sample. Finally, the quantitative and qualitative evaluations of the methods applied in this methodology are reported.

A. Deep learning model

Two different models are chosen to be investigated for detecting the abnormal samples in the collected dataset. The first model is a Fully Convolutional Network (FCN) which consists of convolutional layers. We implemented the same network architecture in [12]. The original FCN is for univariate time series classification. However, the extended FCN model for multivariate time series classification is provided by stacking different time series into channels.

The second model is a recurrent neural network model; the Long Short-Term Memory (LSTM) is used as the recurrent layer. This model is also widely used for time series classification since it explicitly models the temporal relationship in time series data.

These models are separately trained on the training sets, and the weights and biases are stored to be later used for XAI methods. Finally, the DNN model with the highest performance is selected as the final fault detector. Furthermore, using the XAI methods, the main contributors of different features and specific parts of them can be emphasized, adding insight to the fault detection process.

B. XAI methods

This subsection introduces five different XAI methods used in this paper. These XAI methods are chosen since they are suitable for multivariate time series classification and are model agnostic.

1) Occlusion: It is a simple perturbation-based approach. The idea is to replace the input time series with a given baseline (zero baselines are used in our case) and then compute the difference in output. The higher difference in output then represents higher attribution from this part of the time series. The attribution is calculated as follows:

$$\phi(f, x_i) = f(x) - f(x[x_i = 0]) \tag{1}$$

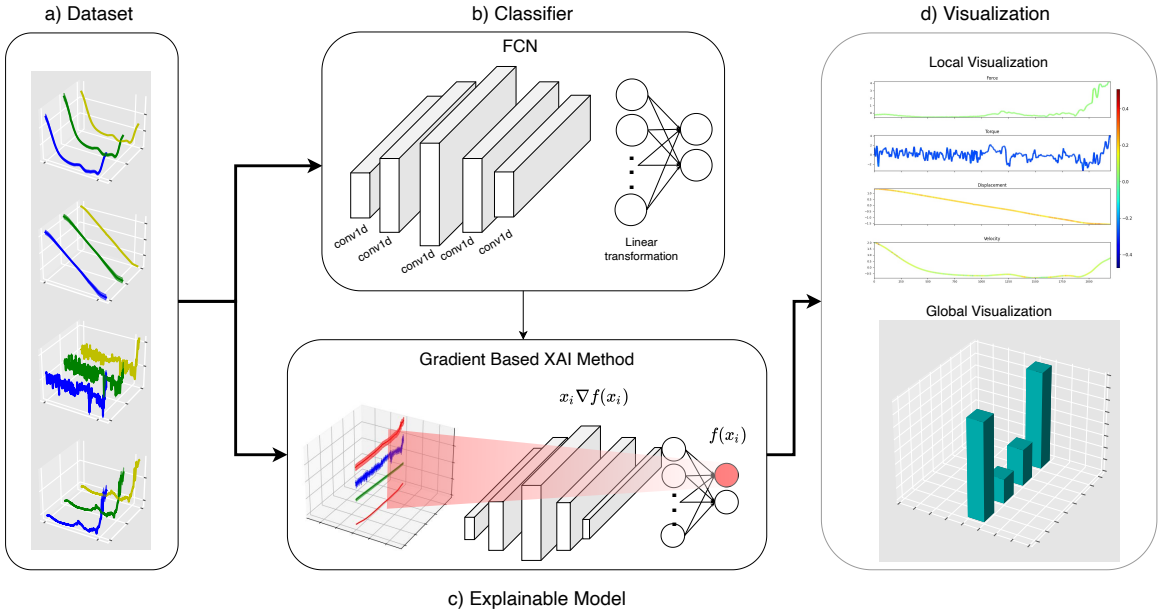

Fig. 1: The overall methodology diagram.

where f is the scoring function (a neural network function). x is the input and $x[x_i = 0]$ denotes that x_i in x is replaced by 0. The method is computationally expensive since it requires multiple forward passes of the network models.

2) Gradient: It returns the gradients with respect to inputs. By definition, it computes the partial derivatives of the target output with respect to each input feature. Here we calculate the global attribution as defined in [17], which is also known as input*gradient in the literature:

$$\phi(f, x_i) = x_i \frac{\partial f(x)}{\partial x_i} \tag{2}$$

3) Integrated gradient: Integrated Gradients [11] is an axiomatic XAI algorithm that assigns an importance score to each input feature by approximating the integral of gradients of the model's output with respect to the inputs along the path from given baselines to inputs:

$$\phi(f, x_i) = (x_i - \bar{x}_i) \int_{\alpha=0}^{1} \frac{\partial f(\tilde{x})}{\partial \tilde{x}_i} \mid_{\tilde{x}=\bar{x}+\alpha(x-\bar{x})} d\alpha \tag{3}$$

where $\bar{x}$ is the given baseline and zero baselines are used in this paper. To integrate the above equation, we discretize α into 50 equally spaced bins for numerical integration.

4) DeepLiftShap and KernelShap: DeepLiftShap and KernelShap are two XAI methods proposed in [16]. These two methods try to approximate the Shapley values in a more efficient way.

DeepLiftShap extends the DeepLift algorithm [18] to approximate the Shapley values. For each input sample, it computes DeepLift attribution with respect to each baseline and averages resulting attributions.

KernelShap uses the LIME framework [19] to compute Shapley Values. By setting the loss function, weighting kernel, and regularization terms appropriately in the LIME framework, the Shapley values can be theoretically derived.

C. Evaluation methods

In order to evaluate the explanations, qualitative and quantitative evaluation methods are used in this paper. Note that the qualitative method measures how the explanation matches the expert's opinion, while the quantitative method indicates how well the explanations satisfy the completeness and sensitivity axioms.

1) Qualitative evaluation: We first ask experts to assign each sensor a relevance score on faults to assess the explanations qualitatively. Then the XAI methods presented in Section III-B are used to provide attribution maps (local explanations) for all the samples in the dataset. The attribution map for each sample is then summed in the time axis, which provides the sensor attribution for each sample. Next, the sensor attributions are averaged across all samples to obtain a global sensor attribution. Finally, the relative cosine similarity score is calculated to measure the similarity between the expert's relevance score and the global sensor attribution from XAI methods:

$$score = \frac{cosine(R_{exp}, R_{attr}) - cosine(R_{exp}, R_{base})}{1 - cosine(R_{exp}, R_{base})} \tag{4}$$

where $cosine$ is the cosine similarity function. It is a measure of similarity between two non-zero vectors and it is defined as the cosine of the angle between the vectors; that is, it is the dot product of the vectors divided by the product of

their lengths. R_{exp} is the expert's relevance score. R_{attr} is the global sensor attribution from XAI methods. R_{base} is the base sensor attribution that all the sensor has equal attribution. Note that the positive score suggests that the global sensor attribution is better than the base one.

2) Quantitative evaluation: Two metrics, infidelity and sensitivity [9], are used for quantitative evaluation. The infidelity measure is derived from the completeness axiom, which is defined as the expected difference between the dot product of the input perturbation to the explanation and the output perturbation:

$$infd(\phi, f, x) = E_I[(I^T\phi(f, x) - (f(x) - f(x - I)))^2] \quad (5)$$

where ϕ is the XAI attribution method. f is the deep learning model. x is input and I is a small perturbation.

Sensitivity measures the extent of explanation change by insignificant perturbations from the test point. It is natural to consider the explanation to have low sensitivity. The sensitivity is defined as follows:

$$sens(\phi, f, x, r) = \max_{||\delta|| \leq r} ||\phi(f, x + \delta) - \phi(f, x)|| \quad (6)$$

where r is a given input neighborhood radius.

D. Visualization

In this paper, two different visualization methods are used to show global and local attribution. First, a bar chart shows the global attribution input features, where each bar represents the global attribution value for each feature. However, to show the local attribution, the attribution value for each data point in different features is displayed in color. Therefore the importance of each input sample can be derived from the color used. The color bar in this type of visualization demonstrates the strength of each color. The overall diagram of the applied methodology is presented in Fig. 1.

IV. Experiments

Two different case studies are investigated to evaluate the proposed methodology: a vessel engine and a medical assembly process. Moreover, the results of implementing different classifiers and explainability algorithms are presented in order. These results, including qualitative and quantitative, are then compared and discussed. Ultimately the derived results show that the methodology can be applied to various industrial use cases of similar nature.

A. Case 1: Vessel engine

1) Dataset: The dataset [20] is collected from a vessel engine in the hybrid power lab, as shown in Fig. 2. In the experiment, the engine aims to replicate real-life ferry crossings. Similar to [20], only nine sensors closely related to the engine's status are used. The train and test data are split by different runs to avoid data leakage.

Two artificial faults, malfunction of the turbocharge and the clogged air filter (Fig. 2), are introduced to the engine during the processes. The turbocharge malfunction is implemented by installing a bleed device on the charge air pipe between the turbocharger and the engine inlet manifold and then gradually bleeding of air. The clogged air filter is simulated by a restriction device, which is gradually adjusted to reduce the inlet flow of air to the turbocharger. Although there are two different faults, the currently installed sensors give almost identical fault symptoms. Therefore fault detection is only treated as a binary classification problem.

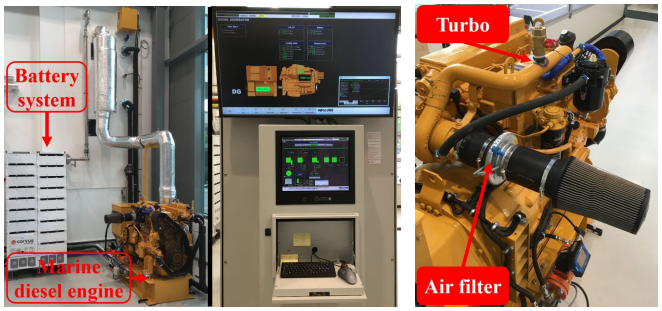

Fig. 2: Left: Battery system, the vessel engine, and the automation system used for collecting the dataset. Right: Restriction and bleed device used to provoke the air filter and turbo fault, respectively. [20]

TABLE I: Performance evaluation of the FCN and LSTM models for vessel engine.

Model	Precision	Recall	F1-score
FCN	0.93	0.99	0.96
LSTM	0.96	0.99	0.98

TABLE II: Qualitative evaluation of different attribution methods for vessel engine fault detection.

Model/XAI	FCN	LSTM	Avg.
Occlusion	**0.350**	-0.224	0.063
Gradient	0.310	0.076	0.193
Integrated Gradient	0.238	**0.214**	**0.226**
DeepLiftShap	0.226	-0.223	0.002
KernelShap	0.239	0.012	0.126

2) Experimental results: The performance of the FCN and LSTM models are evaluated as shown in Table I. The micro-average precision, recall, and F1-score are reported. The results show that both models perform well. However, the LSTM model performs slightly better than the FCN model in terms of precision and F1-score.

Regarding qualitative evaluation, Fig. 3 presents the relevance score provided by the expert and the global sensor attribution calculated from different XAI methods for these two models. The results show that different XAI methods provide similar sensory attribution to the FCN model, highlighting the importance of the boost pressure. For LSTM models, boost pressure and cooling water temperature are highlighted for all XAI methods except KernelShap. The KernelShap attribute almost equally to all sensors. Table II summarizes the relative cosine similarity between the expert's relevance score and the sensor attributions from XAI methods. Although the LSTM performs slightly better than FCN in terms of precision and F1-

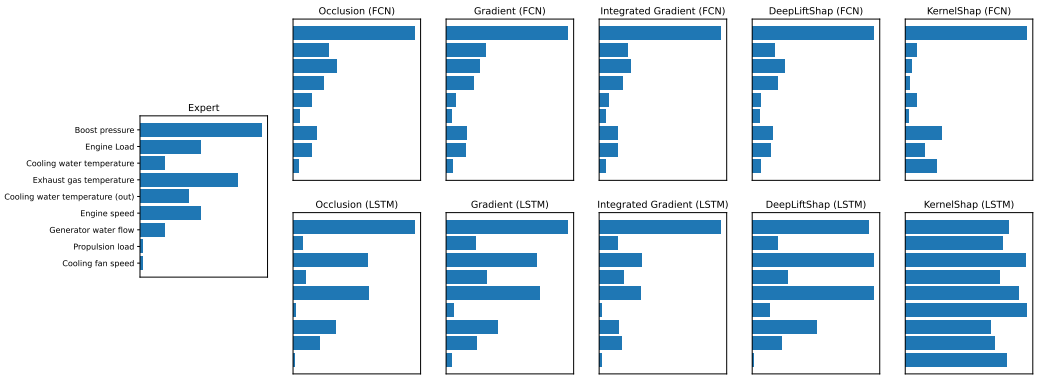

Fig. 3: Relevance score provided by the expert and the global sensor attribution calculated from different XAI methods for the vessel engine.

score, the sensor attributions from FCN are more in line with expert opinion than LSTM (Occlusion has highest similarity score as shown in Table II). The integrated gradient provides consistent results on both FCN and LSTM, and in this case, best match expert opinion due to higher average similarity score.

Regarding quantitative evaluation, Table III shows the infidelity and sensitivity measures for different XAI methods on FCN and LSTM. It is shown that DeepLiftShap provides minor infidelity and sensitivity. However, there is no significant difference between different XAI methods except for KernelShap, whose infidelity is difficult to calculate and whose sensitivity is relatively high. Therefore, ensemble gradients are chosen to explain network predictions and generate heatmaps for visualization for engineers.

Fig. 4 presents a heatmap from the integrated gradient on FCN for a randomly selected fault sample. The model successfully detects the fault, and the integrated gradient highlights the part of the sensor measurements in this time window responsible for the fault. It is shown that the XAI method highlights the drop-down boost pressure, which indicates there might be an air filter or turbo fault.

B. Case 2: Medical device assembly

1) Physical system: The medical device assembly use case is a real-world pharmaceutical manufacturing pilot line with different processes. The process focuses on a snap process in this paper, where two different medical modules called sub-assemblies are mounted together with an applied force and vertical displacement. In addition, the process is equipped with force and torque transducers for continuous process monitoring. The physical system is shown Fig. 5.

2) Dataset: The most relevant measurements for the snap process are Force, Torque, Displacement, and Velocity. We run the process with the normal setting to collect the normal data samples. However, to collect the abnormal samples, two different kinds of abnormality have been introduced in the process. The first type of fault is changing the gripper offset to differ from the calibrated working point. While in the second type of fault, we change the structure of the sub-assembly. We remove some deformation structures in the products. The combination of normal and abnormal samples is split into training and test sets with the ratio of $[80\%, 20\%]$ to train the DNN model.

3) Experimental results: Two DNN models are designed to detect the abnormal samples in this use case, FCN, and

TABLE III: Quantitative evaluation of different attribution methods for vessel engine fault detection. (infiedilty/sensitivity-max)

Model/XAI	FCN	LSTM	Avg.
Occlusion	4.112 / **0.033**	5.428 / 0.023	4.770 / 0.028
Gradient	4.911 / 0.086	3.812 / 0.066	4.362 / 0.076
Integrated Gradient	4.642 / **0.033**	2.275 / 0.031	3.459 / 0.032
DeepLiftShap	**3.277** / 0.034	**0.117** / **0.018**	**1.697** / **0.026**
KernelShap	- / 1.288	- / 1.387	- / 1.338

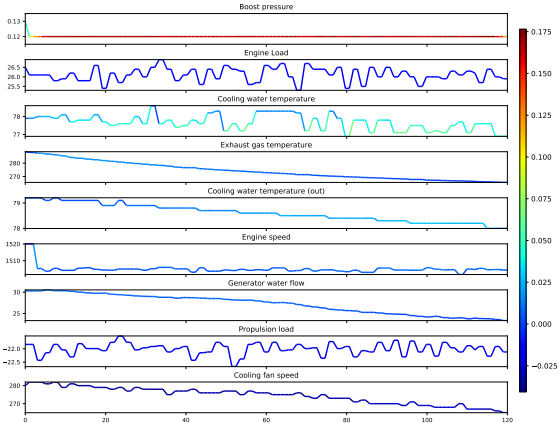

Fig. 4: Heatmap from integrated gradient on FCN for a selected fault sample.

TABLE IV: Performance evaluation of the FCN and LSTM models for the medical device assembly.

Model	Precision	Recall	F1-score
FCN	1	1	1
LSTM	0.954	0.954	0.954

LSTM. Furthermore, different attribution methods have been applied to explain the contribution of different measurements. The classification performance results are shown in Table IV. The results show that FCN is performing slightly better than LSTM.

The qualitative results derived from different attribution methods for FCN and LSTM models are shown in Fig. 6. The attribution distribution for the FCN model in general show better alignment with the expert expectation. Furthermore, by calculating the similarity between the relevance score provided by the expert and the global sensor attribution calculated from different XAI methods, shown in Table V, we can conclude that the DeepLiftShap and the FCN model is the best performing XAI algorithm that matches the expert opinion.

Furthermore, the quantitative results, including the Infidelity and Sensitivity of the attributions, are mentioned in Table VI. The average value of these criteria are calculated for various samples. According to these results the lowest infidelity belong to the Gradient method in FCN and DeepLiftShap in LSTM. Furthermore, the lowest sensitive algorithm is Occlusion in FCN and LSTM.

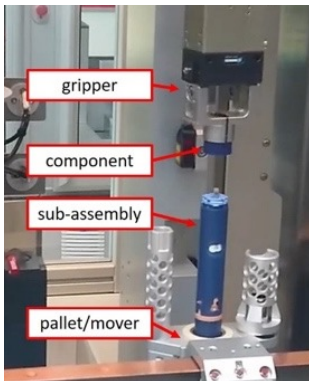

Fig. 5: Illustration of the medical device assembly machine. In this figure a gripper, mounted on linear motor holds a component which must be mounted to a sub-assembly for the fabrication of a device. The transport system moves a pallet, also called mover, which contains components to be assembled. [21]

.

TABLE V: Qualitative evaluation of different attribution methods for the medical device assembly fault detection.

Model/XAI	FCN	LSTM	Avg.
Occlusion	−1.878	−4.568	−3.224
Gradient	0.426	−0.645	−0.1095
Integrated Gradient	0.601	−0.6327	−0.0158
DeepLiftShap	**0.91**	−0.645	**0.133**

C. Discussion

The qualitative and quantitative results demonstrate that different XAI algorithms and classifier models perform differently. For example, in the vessel engine use case, the FCN model and Occlusion have better qualitative performance and are more aligned with expert opinion; however, in total integrated gradient is the method with the highest average performance for both FCN and LSTM models. Furthermore, the quantitative evaluation of XAI methods emphasizes that DeepLiftShap has the lowest infidelity, and on average the lowest sensitivity.

On the other hand, in the medical device assembly use case, DeepLiftShap, together with the FCN classifier, has better qualitative results, which ensure better alignment with the expert expectation. However, quantitative results in the same use case favor lower infidelity and sensitivity, which results in the DeepLisftShap method and the LSTM classifier for lowest infidelity and Occlusion for the lowest sensitivity.

The final results in this paper show that we cannot choose a final classifier and XAI algorithm that perform best in both qualitative and quantitative evaluations. Since these criteria do not always agree, sometimes the combination of a classifier and XAI algorithm performs well regarding qualitative evaluation and is aligned with expert knowledge. However, the same combination can have a low performance in terms of infidelity and sensitivity. Therefore, choosing the final interpretable fault detector, one must make a compromise between qualitative and quantitative results.

V. Conclusions

In this paper, we examine five different XAI algorithms for deep neural networks in two separate but relatively sensitive fault detection cases: vessel engine and an industrial, medical device assembly. In addition, two different neural network models widely used for time series classification are employed. The selected network models and XAI algorithms are universal and can be applied to various fault detection cases.

Findings suggest that a network model with a specific XAI algorithm that agrees with the expert knowledge might not exhibit lower infidelity or sensitivity and vice versa. Therefore, choosing the XAI algorithm with a network model for a specific fault detection case compromises expert knowledge and quantitative results.

Acknowledgment

We would like to thank the following people from Novo Nordisk A/S; Thomas Algot Søllested, Tinna Dofradóttir,

TABLE VI: Quantitative evaluation of different attribution methods for the medical device assembly. (infiedilty/sensitivity-max)

Model/XAI	FCN	LSTM	Avg
Occlusion	7.863 / **0.1092**	6.5209 / **0.0283**	7.192 / **0.06875**
Gradient	**2.3598** / 0.405	2.861 / 0.0599	2.6104 / 0.232
Integrated Gradient	2.5674 / 0.4442	3.192 / 0.3629	2.8797 / 0.4035
DeepLiftShap	2.7029 / 1.0573	**1.9121** / 0.3178	**2.3075** / 0.6876

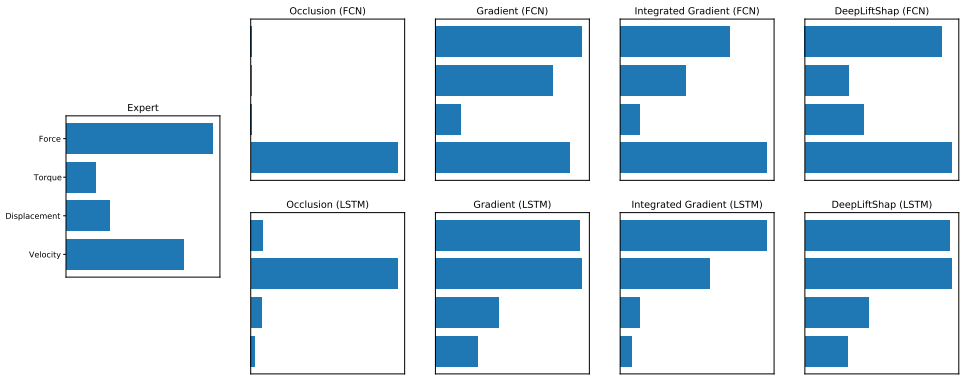

Fig. 6: Relevance score provided by the expert and the global sensor attribution calculated from different XAI methods for the medical device assembly

and Sebastian Dengler for providing the physical system, conducting data collection, and insights into understanding the assembly process. Additionally, we extend our gratitude to Innovation Foundation Denmark for funding the MADE FAST project, of which this work is a part. Finally, we would like to acknowledge the insights provided by Finn Tore Holmeset on the vessel engine.

REFERENCES

[1] F. Kakavandi, R. de Reus, C. Gomes, N. Heidari, A. Iosifidis, and P. G. Larsen, "Product quality control in assembly machine under data restricted settings," in *IEEE 20th International Conference on Industrial Informatics (INDIN)*, 2022, pp. 735–741.

[2] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *Journal of Manufacturing Systems*, vol. 48, pp. 144–156, 2018, special Issue on Smart Manufacturing.

[3] P. V. Henstock, "Artificial intelligence for pharma: time for internal investment," *Trends in pharmacological sciences*, vol. 40, no. 8, pp. 543–546, 2019.

[4] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information Fusion*, vol. 58, pp. 82–115, 2020.

[5] A. Adadi and M. Berrada, "Peeking inside the black-box: a survey on explainable artificial intelligence (xai)," *IEEE access*, vol. 6, pp. 52 138–52 160, 2018.

[6] A. Das and P. Rad, "Opportunities and challenges in explainable artificial intelligence (xai): A survey," *arXiv preprint arXiv:2006.11371*, 2020.

[7] T. Rojat, R. Puget, D. Filliat, J. Del Ser, R. Gelin, and N. Díaz-Rodríguez, "Explainable artificial intelligence (xai) on timeseries data: A survey," *arXiv preprint arXiv:2104.00950*, 2021.

[8] F. Kakavandi and P. G. Larsen, "Explainable product quality assessment in a medical device assembly pilot line," in *2022 10th International Conference on Control, Mechatronics and Automation (ICCMA)*, 2022, pp. 271–275.

[9] C. K. Yeh, C. Y. Hsieh, A. Suggala, D. I. Inouye, and P. K. Ravikumar, "On the (in) fidelity and sensitivity of explanations," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[10] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[11] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International conference on machine learning*. PMLR, 2017, pp. 3319–3328.

[12] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," in *2017 International joint conference on neural networks (IJCNN)*. IEEE, 2017, pp. 1578–1585.

[13] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2921–2929.

[14] K. Kashiparekh, J. Narwariya, P. Malhotra, L. Vig, and G. Shroff, "Convtimenet: A pre-trained deep convolutional neural network for time series classification," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[15] F. Oviedo, Z. Ren, S. Sun, C. Settens, Z. Liu, N. T. P. Hartono, S. Ramasamy, B. L. DeCost, S. I. Tian, G. Romano *et al.*, "Fast and interpretable classification of small x-ray diffraction datasets using data augmentation and deep neural networks," *npj Computational Materials*, vol. 5, no. 1, pp. 1–9, 2019.

[16] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[17] M. Ancona, E. Ceolini, C. Öztireli, and M. Gross, "Towards better understanding of gradient-based attribution methods for deep neural networks," *arXiv preprint arXiv:1711.06104*, 2017.

[18] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *International conference on machine learning*. PMLR, 2017, pp. 3145–3153.

[19] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[20] A. L. Ellefsen, P. Han, X. Cheng, F. T. Holmeset, V. Æsøy, and H. Zhang, "Online fault detection in autonomous ferries: Using fault-type independent spectral anomaly detection," *IEEE Transactions on instrumentation and measurement*, vol. 69, no. 10, pp. 8216–8225, 2020.

[21] F. Kakavandi, C. Gomes, R. d. Reus, J. Badstue, J. Langdal, P. G. Larsen, and A. Iosifidis, "Towards developing a digital twin for a manufacturing pilot line: An industrial case study," in *Digital Twin Driven Intelligent Systems and Emerging Metaverse*. Springer Nature, in press.