Al-Mukhtar Journal of Engineering Research 09 (1): 47-56, 2025
Doi: https://doi.org/10.54172/2hjmxd48

Research Article **Open Access**

Dynamic Fault Detection in Automotive Engines using Principal Component Analysis based Hybrid Radial Basis Function and Input Training Neural Network Models

Adnan Hamad[1*], Dingli Yu[2]

*Corresponding author: adnan.hamad@omu.edu.ly Department of Electrical Engineering, Faculty of Engineering, Omar Al-Mukhtar University, Libya

[2] Engineering and Technology Department, Liverpool John Moores University, United Kingdom

Received:
20 May 2025

Accepted:
25 December 2025

Publish online:
31 December 2025

Abstract

For automobile engines, a novel fault detection (FD) system is created using PCA in this paper. To identify errors, two distinct neural networks are used. Radial basis function (RBF) is the first neural network, while input training neural network (ITNN) is the second. The mean value engine model (MVEM) with Matlab/Simulink is used to build the approach and evaluate its performance. The MVEM has been used to simulate three faults. The outputs of the MVEM are used as input data for the RBF. ITNN received the RBF output as an input and the output were the estimation of speed, pressure and temperature. According to the simulation results, faults with an amplitude variation of 10 – 20% were successfully identified under dynamic conditions across the whole working range. The corresponding detection thresholds are 0.36, 0.68, and 0.284 for speed, pressure and temperature respectively and any error exceeding the allowable threshold will be easily detected. Therefore, the simulation demonstrates that all three flaws may be easily identified and yields satisfactory results.

Keywords: Principal Component Analysis (PCA), Automotive Engines, Radial Basis Function (RBF) Neural Networks, Fault Detection (FD), Input Training Neural Network (ITNN).

INTRODUCTION

The oldest and most well-known multivariate analysis method is most likely PCA. Pearson was the first to present it (Pearson 1901), and Hotelling developed it on their own (Hotelling 1933). The main goal of PCA is to minimise the dimensionality of a data set with many interconnected variables while preserving as much of the data set's variance as feasible. This is accomplished by converting them into a new set of uncorrelated variables called principal components (PCs), which are arranged so that the majority of the variation found in all of the original variables is retained in the first few (Jolliffe 2002). In recent years, the process industries are aimed to make tighter product specifications and lower production costs. In order to address this subject, process measurements are utilized and manipulated variables collected on the process, many of which exhibit similar behavior due to the underlying physical and chemical relationships, which result in some of the process variables being correlated or collinear. Measurement interdependence is a process feature which is capable of identifying every flaw in the manufacturing process which may cause poor quality production. PCA takes advantage of the correlations between process measurements and can be used. Non-linear PCA technique based on the Input-Training neural network (ITNN) has proposed by Tan and Saif (Tan, et al. 2000) as non-linear models between observed variables and latent variables. The study has shown that the ITNN is equivalent to the principal curve method.. Baligh et al. developed PCA-based fault detection and diagnosis techniques to identify faults in process operation conditions (Mnassri, B. et al. 2009). In order to provide an accurate cause for abnormal process conditions that provide more precise information about the failure cause, the suggested diagnosis method makes use of two new contribution plots. The findings demonstrate how simple it would be to arrive at an accurate conclusion using these techniques. In order to fully utilise the inherent parallelism of high-performance computing platforms and, consequently, shorten the processing time of a given hyperspectral image, Martel, E. and et al (2018) presented the implementation of the PCA algorithm onto two distinct high-performance devices, namely an NVIDIA Graphics Processing Unit (GPU) and a Kalray many-core. For the first time, a thorough analysis highlighting the advantages and disadvantages of each option has been provided by comparing the obtained results with various hyperspectral images with those obtained with a field programmable gate array (FPGA)-based implementation of the PCA algorithm. Also, a dynamic principal component analysis (DPCA) method was proposed by Lin, C et al (Lin et al. 2022). They suggested a defect diagnosis and monitoring technique that takes into account the dynamic properties of multivariate data and minimises the dimension of the input matrix by combining feature selection with DPCA. The study demonstrates that the suggested approach significantly increases accuracy. In a sizable, broadly applicable dataset of people with bipolar illnesses and controls, principal component analysis (PCA) is used by McWhinney, S. et al (2024) to find patterns of covariance across brain areas and link them to clinical and demographic characteristics.

Konishi, T. (2025) employed PCA to extract dominant patterns from multivariate data and demonstrated that slight rotations of the principal components can significantly improve result interpretability. Their findings showed that PCA effectively distinguishes underlying data behaviors, supporting its suitability for fault detection applications where clear separation between normal and abnormal conditions is required.

In this paper ITNN and RBFNN are utilised to detect simulated problems. this sort of neural networks has been chosen because of has a simple structure and that it is easy to train. Also, the primary characteristic of ITNN is that, using the steepest gradient descent network optimisation rule, each input pattern is modified in conjunction with internal network parameters to minimise the matching output pattern (Jia et al. 2000). Therefore, by combining RBFNN and ITNN and leveraging the strengths of each technique, improved overall performance and more results are expected.

STRATEGIES AND METHODS

Input training neural network (ITNN)

Figure 1 shows an illustration of the ITNN's architecture (Dong et al. 1996). It consists of three layers: input, output, and one hidden layer. The input layer has nodes equivalent to the required reduced dimension of the data. The hidden layer facilitates the extraction of the non-linear mapping between the output variables and the reduced set. Nodes' input and output layers could have a non-linear activation function or an identity. To capture non-linear correlation between the variable in the data set and the reduced set, which is non-linear principal components, the hidden layer of nodes needs to contain a non-linear activation function, such as sigmoid, tanh, or logistic. (Dong et al. 1996). The ITNN has a unique property that is during the training, every input vector is modified to reduce the error between the original sample and the matching ITNN output. The calculation of the ITNN input layer gives the ITNN an exclusive property which makes it different with a regular neural network. Random input values (network parameters) were used at the beginning in order to train the ITNN (Malthouse et al. 1997). The training continued until we defined the optimal input values and internal network parameters. Each input vector can be considered as a non-linear principal component vector, and it is associated with the corresponding output vector which is a process measurement vector. For quality result, the input vector adjusted to minimize the output vector error (Reddy et al. 1998).

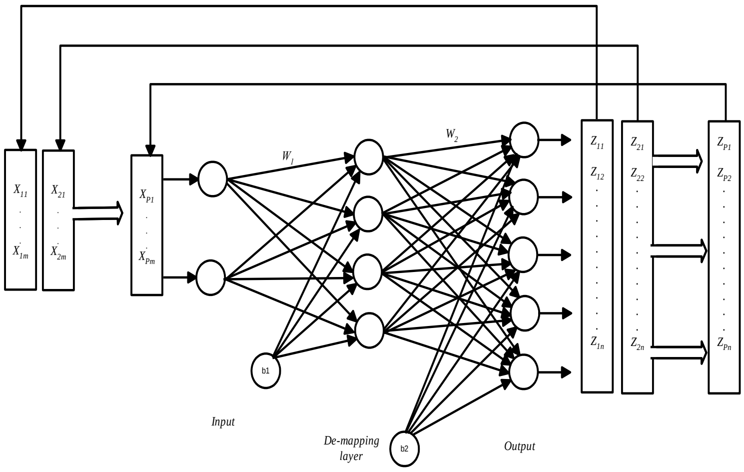

Figure: (1). The structure of ITNN

Fault Detection Approach Based Upon Non-Linear PCA for Automotive Engine

Collecting data

To evaluate the performance of various adaptive engine models under real-world driving conditions, the training data used in engine data collection must be reflective of typical plant behaviour. Accordingly, input and output signals must sufficiently span the area that the system will be monitored (Draeger et al. 1995). The throttle angle and fuel injection rate are the two inputs used in the engine model. A series of random ambulated signals (RAS) were created for fuel injection and throttle angle between 0.0009 kg/s to 0.0018 kg/s and 30 and 50 degrees respectively in order to obtain the engine data for ITNN training. The data collected from MVEM (Hendricks et al. 2000) are speed, pressure and temperature. These data were used as a target matrix for the IT neural network (Hamad et al. 2010, 2011 and 2012) (see Figure 2).

Train the Model Using ITNN and RBFNN

Figure 2 shows the block diagram of train the ITNN and RBFNN using engine data. The output of MVEM (speed, pressure and temperature) was used as a target matrix for the IT neural network.

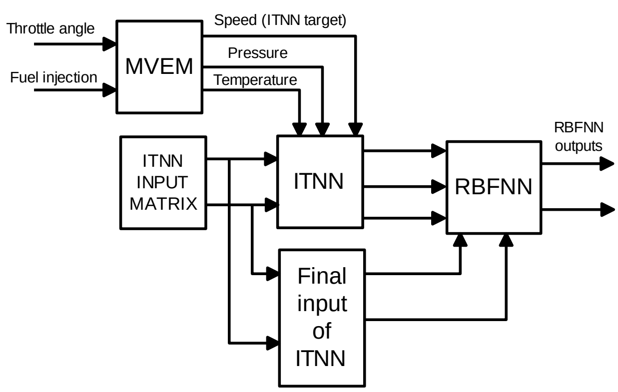

Figure: (2). Block diagram for train the ITNN and RBFNN based on non-linear P CA using MVEM data

6000 samples were used. The output of ITNN after training was matrix with dimensions 3 rows and 6000 columns, this matrix is the estimation of speed pressure and temperature. Non-linear PCA scores (NLPCAS) were obtained from the final ITNN input matrix after training was finished. Non-linear PCA scores matrix has 2 rows and 6000 columns; this matrix was used as a target matrix for RBFNN. In order to train RBFNN, the ITNN output was used as an input matrix for RBFNN. Figure 3 illustrates the flowchart of the training approach. 12 hidden nodes are chosen by k-means method. The output of trained RBFNN is matrix with 6000 columns and 2 rows. After obtaining trained ITNN and trained RBFNN, the model in Figure 2 can be used to detect the fault in the automotive engine. This model will be used for the following two cases. The first case is using engine data without fault, and the second case is using faulty engine data. Then use the faulty data in order to see how well the validity of the model to detect the fault for automotive engine.

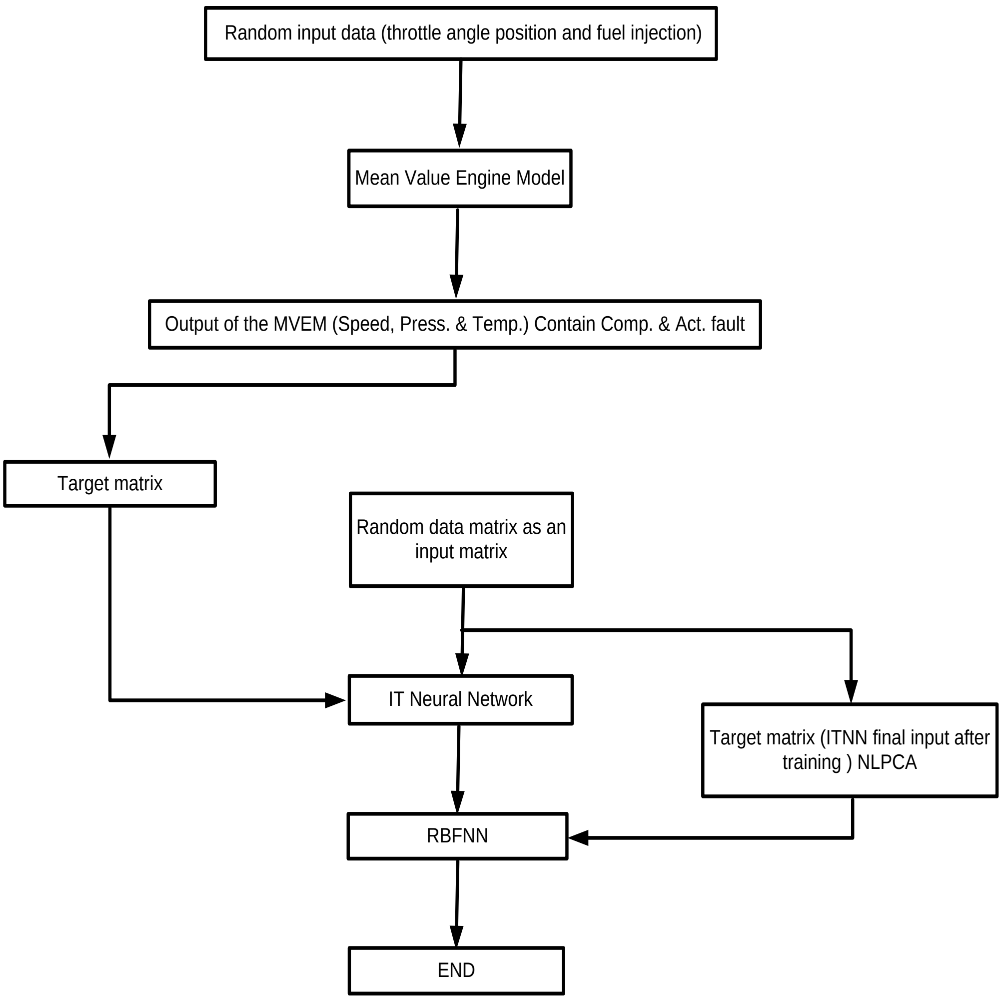

Figure:(3). The flow chart of train the ITNN and RBFNN using engine data

Fault Simulation

A bias equal to 10–20% of the sensor's typical output is applied to simulate failures with speed sensors, pressure and temperature. From samples 2001 to 2500, 3501 to 4000 and 4001 to 4500, respectively (see Figure 4), these faults are simulated. 1.1 and 1.2 multiplying factors (MFs) are applied to create the sensor's inaccurate data as shown in Table 1. For every fault state, the updated MVEM generates faulty data with throt multiplication factors (MFs). Practically every transient condition of the engine dynamics is covered by the engine data for the simulated faults.

RESULTS

Detecting Faults using the PCA Model

The block diagram for fault detection utilising the ITNN/RBFNN model is displayed in Figure 5. The RBF neural network contains two outputs that each indicate one of the states under investigation in Table 1. The network gets three input signals—manifold pressure, temperature, and crankshaft speed—all of which contain inaccurate information (Hamad et al. 2010, 2011 and 2012). Trained ITNN will receive the RBFNN outputs as an input in order to estimate the crankshaft speed, temperature and manifold pressure. The test results for fault identification prior to filtering are displayed in Figures 6–8, and it is evident that the errors are significant where the x axis represents the number of data samples, while the y axis represents the error value . Figures 9–11 show the excellent outcomes and reduced errors following the filtering operation. For the crankshaft speed, manifold pressure, and manifold temperature error signals, the corresponding detection thresholds are 0.36, 0.68, and 0.284. Low thresholds will result in false alarms, while high thresholds could miss detections. The selection of thresholds is based on experience in reducing the false alarm rate.

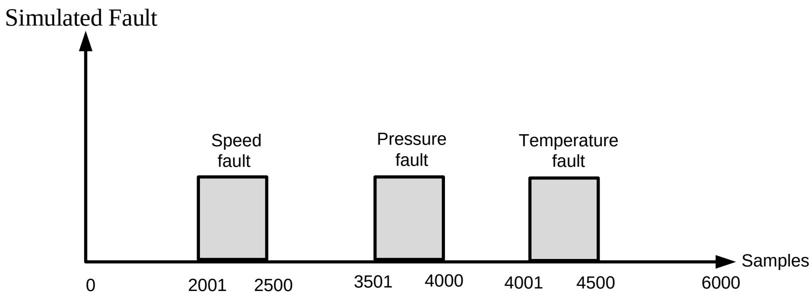

Figure:(4). The simulated faults

Table:(1). Multiplying factors and the three fault states

No.	Fault	Multiplication Factors
1	20% over-reading on the output of the speed sensor	1.2
2	20% over-reading on the output of the Pressure sensor	1.2
3	10% over-reading on the output of the Temperature sensor	1.1

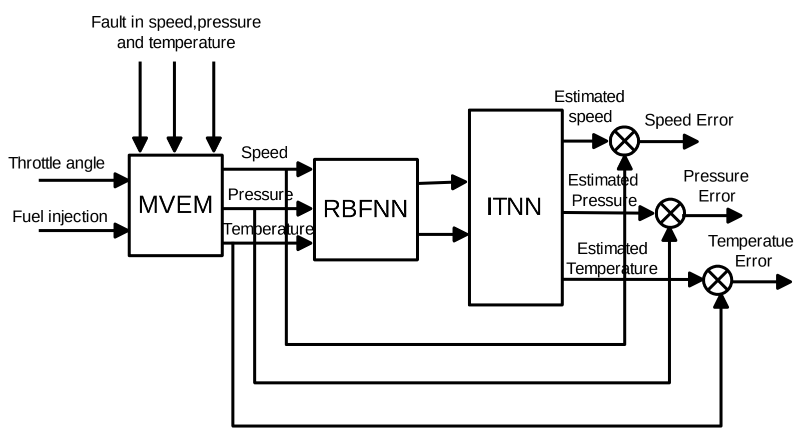

Figure:(5). Block diagram for testing the model based on non-linear PCA using MVEM data with faults

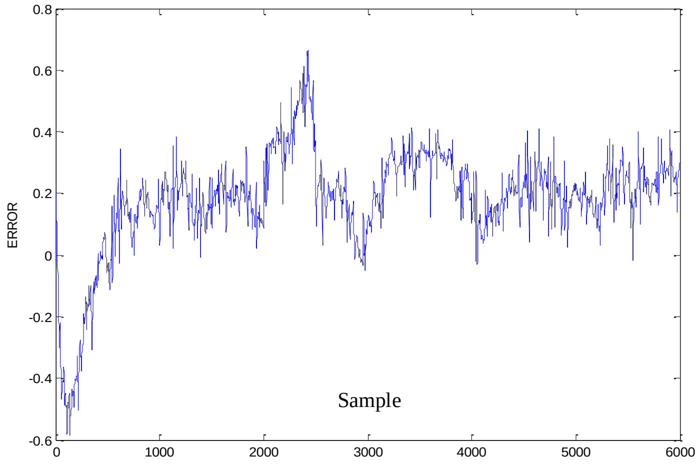

Figure:(6). The error signal between the RBFNN speed input and ITNN speed output in case data engine with faults before the filtering

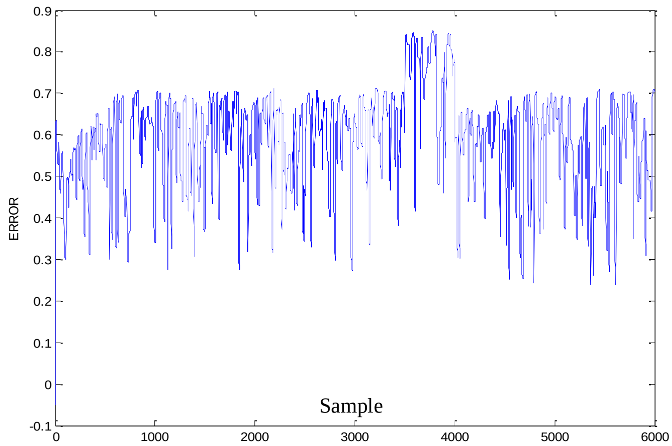

Figure:(7). The error signal between the RBFNN pressure input and ITNN pressure output in case data engine with faults before the filtering

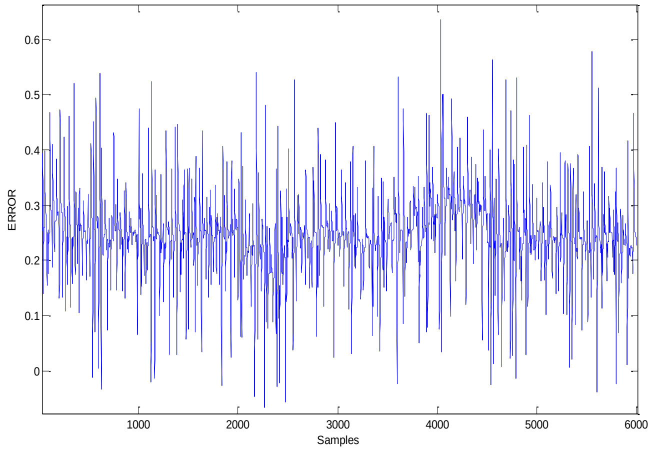

Figure:(8). The error signal between the RBFNN temperature input and ITNN temperature output in case data engine with faults before the filtering

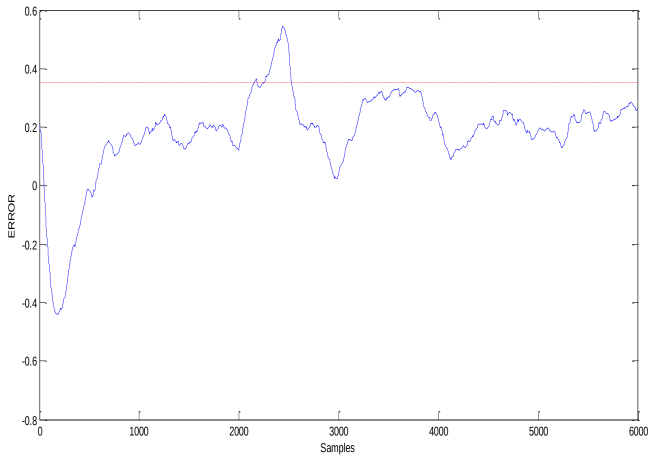

Figure:(9). The error signal between the RBFNN speed input and ITNN speed output in case data engine with faults after the filtering

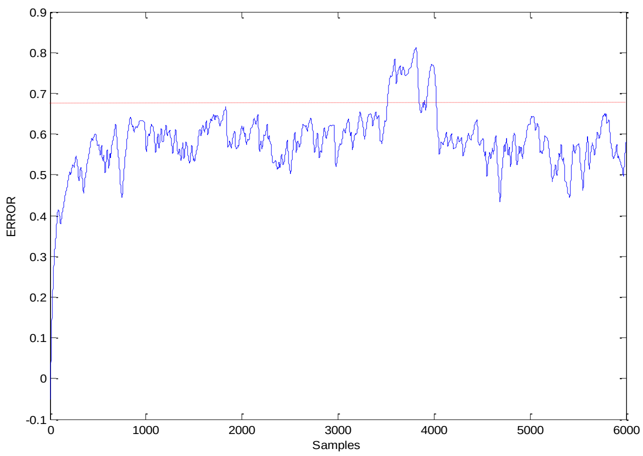

Figure: (10). The error signal between the RBFNN pressure input and ITNN pressure output in case data engine with faults after the filtering

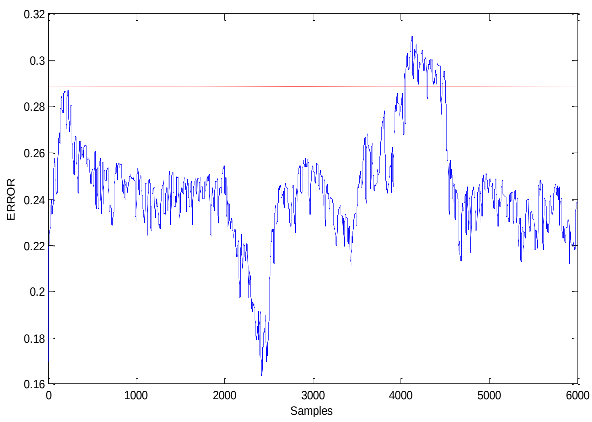

Figure:(11). The error signal between the RBFNN temperature input and ITNN temperature output in case data engine with faults after the filtering

DISCUSSION

The test results for defect identification prior to filtering are displayed in Figures 6–8, where each figure display the error signal between the RBFNN (speed, pressure or temperature) input and ITNN (speed, pressure or temperature) output in case data engine with faults.while the test results following filtering are displayed in Figures 9–11 (which are the figures 6-8 after filtering). It is evident from these numbers that every flaw was found with great clarity. For scaled data, the specified detection thresholds were 0.36 for crankshaft speed, 0.68 for manifold pressure, and 0.284 for manifold temperature.

CONCLUSION

During the research period, simulations are conducted using the MVEM developed by Hendricks et al. (Hendricks et al. 2000) with minor modifications. The MVEM simulation has been expanded to include air fuel ratio sensor time delay, temperature sensor dynamics, and other features. Three faults (speed, pressure, and temperature faults) are simulated, and the faults are detected using two distinct neural networks (ITNN and RBF). The MVEM output was sent into the first neural network, the RBF neural network, which produced the estimation matrix for the non-linear PCA. ITNN was the second neural network. This neural received the RBFNN output as an input and the output was the estimation of speed, pressure and temperature. This operation was done in two cases, data without faults and with faulty data. The faults of 10-20 % amplitude were successfully detected. The MVEM, an engine simulation benchmark, is used to model the flaws, and the created approach is assessed. Satisfactory results are achieved in simulation, and the simulation shows that all three faults can be clearly detected.

Duality of interest: The authors of this manuscript declare that they have no conflicting interests.

Author contributions: Adnan Hamad: Writing – original draft, design of methodology, develop the software and supporting algorithms, Verification, data collection, investigation process. Dingli Yu: writing- review and editing, Conceptualization, research data, implementation of the computer code(supporting).

Funding: No specific grant from any financial organisation was obtained for this study.

REFERENCES

Dong, D. and McAvoy, T. J. (1996). Non-linear principal component analysis based on principal curve and neural networks, *Comput Chem, Eng*, 20: 65-78.

Draeger, A. Engell, S. and Ranke, H. (1995). Model predictive control using neural networks. *IEEE Control Systems Magazine*, 15(5):61-66.

Hamad, A., Yu, D. L., Gomm, J. B. and Sangha, M. (2010). Automotive engine fault detection and isolation via radial basis function neural network. Proceeding Of 16th International Conference On Automation & Computing, *University of Birmingham, UK*, pp. 232-237 , ISBN978-0-9555293-6-8. 163

Hamad, A., Yu, D. L., Gomm, J. B. and Sangha, M. (2010). Radial basis function neural network in fault detection of automotive engines. *International Journal of Engineering, Science and Technology*, Vol. 2, No. 10, pp. 1-8.

Hamad, A., Yu, D. L., Gomm, J. B. and Sangha, M. (2010). Automotive engine fault detection via radial basis function neural network. *The 6th GERI Annual Research Symposium* (GARS 2010) / Friday 25th June 2010 at Liverpool john Moore University.

Hamad, A., Yu, D. L., Gomm, J. B. and Sangha, M. (2011). Fault diagnosis for engine air path with neural models and classifier. *International Journal of Engineering, Science and Technology*, Vol. 3, No. 8, pp. 66-86. www.ijest-ng.com

Hamad, A., Jones, K., Yu, D. L., Gomm, J. B. and Sangha, M. (2011). Residual generation for fault detection of internal combustion engine. *International Conference on Computer Systems and Technologies*- CompSysTech'11, 17 June 2011, Vienna. pp. 329-334

Hamad, A., Yu, D. L., Gomm, J. B. and Sangha, M. (2012). Dynamic fault detection and isolation for automotive engine air path by independent neural network model. *International Journal of Engine Research*. 0(0) 1–14 IMechE 2012. DOI: 10.1177/1468087412461267 jer.sagepub.com

Hamad, A., Yu, D. L., Gomm, J. B. and Sangha, M. (2012). Fault detection and isolation for engine under closed-loop control. *UKACC International Conference on Control*, Cardiff, UK, 3-5 September, pp. 431-436.

Hendricks, E., Engler, D. and Fam M. (2000), A Generic Mean Value Engine Model for Spark Ignition Engines, *In proceeding of 41st simulation conference*, Denmark. DTU Lyngby,2000.

Hotelling, H. (1933). Analysis of a Complex of Statistical Variables into Principal Components, *Journal of Educational Psychology*, 24(6 & 7), 417–441 & 498–520.

Jia, F. Martin, E. B. and Morris, A. J. (2000) Non-linear principal components analysis with application to process fault detection, *International Journal of Systems Science*, pp. 1473-1487.

Jolliffe, I. T. (2002) Principal Component Analysis. Second ed. Springer Series in Statistics. New York: Springer-Verlag New York.

Konishi, T. (2025). Means and Issues for Adjusting Principal Component Analysis Results, *Algorithms* 2025, 18(3), 129; https://doi.org/10.3390/a18030129

Lin, C., Bai, J. and Wu, F. (2022). Fault Diagnosis Using Dynamic Principal Component Analysis and GA Feature Selection Modelling for Industrial Processes. Processes, 10(12), 2570; https://doi.org/10.3390/pr10122570

Malthouse, E. C. Mah, R. S. and Tamhane, A. C. (1997). Non-linear partial least square, Comput Chem. Eng, 21: 875-889.

Martel, E. , Lazcano,R., López, J., Madroñal, D., Salvador, R., López, S., Juarez, E., Guerra, R., Sanz, C., and Sarmiento, R. (2018). Implementation of the Principal Component Analysis onto High-Performance Computer Facilities for Hyperspectral Dimensionality Reduction: Results and Comparisons, Remote Sens. 2, 10, 864; doi:10.3390/rs10060864

McWhiney, S., Hlinka, J., Bakstein, E., Dietze, L., Corkum, E., (2024). Principal component analysis as an efficient method for capturing multivariate brain signatures of complex disorders—

ENIGMA study in people with bipolar disorders and obesity. 2;45(8):e26682. doi: 10.1002/hbm.26682

Mnassri, B., El Adel, M., Ananou, B. and Ouladsine, M. (2009). Fault Detection and Diagnosis Based on PCA and a New Contribution Plots. Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes Barcelona, Spain, June 30 - July 3, 2009

Pearson, K. (1901). On Lines and Planes of Closest Fit to Systems of Points in Space, The London, Edinburgh, and *Dublin Philosophical Magazine and Journal of Science*, Series 6, 2(11), 559-572.

Reddy, V. N. and Mavrovouniotis, M. L. (1998). An input-training neural network approach for gross error detection and sensor replacement, *Institution of Chemical Engineering*, Trans IChemE, vol 76, part a, pp.478-489.

Tan, Y., Saif, M. (2000). Neural-networks-based nonlinear dynamic modelling for automotive engines, *Neurocomputing*, 30, pp. 129-142.