HYBRID DEEP NEURAL NETWORK MODEL FOR REMAINING USEFUL LIFE ESTIMATION

Ali Al-Dulaimi[†] *and Soheil Zabihi*[†]*, Amir Asif*[†]*, and Arash Mohammadi*[‡]

[†]Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada H3J 1P8
[‡]Concordia Institute for Information System Engineering (CIISE), Concordia University, Canada

ABSTRACT

The paper proposes a Hybrid Deep Neural Network (HDNN) framework for remaining useful life (RUL) estimation for prognostic health management applications. The proposed HDNN framework is the first hybrid model designed for RUL estimation that integrates two deep learning architectures simultaneously and in a parallel fashion. More specifically, in contrary to the majority of existing data-driven prognostic approaches for RUL estimation, which are developed based on a single deep model and can hardly maintain satisfactory generalization performance across various prognostic scenarios, the proposed HDNN framework consists of two parallel paths (one based on Long Short Term Memory (LSTM) and one based on convolutional neural networks (CNN)) followed by a fully connected multilayer fusion neural network, which acts as the fusion center combining the outputs of the two paths to form the target RUL. The proposed HDNN framework is tested on the NASA commercial modular aero-propulsion system simulation (C-MAPSS) dataset. Our comprehensive experiments and comparisons with several recently proposed RUL estimation methodologies developed based on the same data-sets show that the proposed HDNN framework significantly outperforms all its counterparts in the complicated prognostic scenarios with increased number of operating conditions and fault modes.

Index Terms— Deep Learning, Hybrid models, Remaining Useful Life (RUL), Long Short-Term Memory Network (LSTM), Convolutional Neural Network (CNN), Machine Health Monitoring.

1. INTRODUCTION

Aging critical infrastructures and valuable machineries together with recent catastrophic incidents such as the collapse of Morandi bridge in the Italian city of Genoa, have drifted the signal processing away from pure analysis to call for an urgent quest to design advanced and innovative data-driven solutions and efficiently incorporate multi-sensor streaming data sources for industrial development. Prognostic health management (PHM) is among the most critical disciplines that employs the advancement of the great interdependency between signal processing and machine learning techniques to form a key enabling technology to provide an early warning of failure, in several domains ranging from manufacturing and industrial systems to transportation and aerospace. In this regard, the paper proposes an advanced data-driven and multiple model framework, referred to as the hybrid deep neural network model (HDNN), to accurately estimate the remaining useful life (RUL) of a critical system.

The focus of prognostic is mainly on predicting the residual lifetime during which a device can perform its intended function, i.e., estimating the RUL [1] of a system. The RUL estimation has an important role in different areas including aircraft industries, medical equipment, and power plants, that inspired the researchers to develop a variety of RUL prediction approaches [2–6]. The RUL estimation methodologies can be classified into three main categories: (i) Statistical/Model based approaches; (ii) Data-driven techniques, and; (iii) Hybrid solutions [7]. The first category is the most common method that hardly fit to different prognostic applications. Recently another attractive alternative (Item (ii)) using deep neural network, which is equipped to handle prognostic issues of complex mechanical systems whose degradation processes are difficult to be interrelated by statistical-based solutions. However most of the existing approaches belonging to the second category, only incorporate one single model, which can hardly maintain good generalization performance across various prognostic scenarios, especially when this model is well configured for a certain scenario. The paper addresses this gap and focuses on hybrid (multiple-model) solutions (Item (iii)), which have great potentials to address this issue and significantly improve the RUL estimation accuracy.

Prior Work: Recently, different researchers have demonstrated revolutionary progress of using deep learning models by utilizing variety of deep architectures to tackle different tasks of significant engineering importance within manufacturing and industrial systems. For example, References [8–15] proposed different frameworks based on convolutional neural networks (CNN). In addition to the CNNs, long short term memory (LSTM) [16–18] is another main architecture that is recently utilized in deep learning. All the aforementioned approaches and the majority of existing RUL estimation methods only incorporate a single deep neural network technique. A hybrid approach, on the other hand, has the objective to combine advantages of different techniques through their integration such that the results can be aggregated to improve the prediction performance [19]. Zhang *et al.* [20] built a LSTM recurrent neural network (LSTM-RNN) to capture and learn the long-term dependencies among the degraded capacities of lithium ion batteries. Zhao *et al.* [21] introduced another hybrid model, where a deep neural network (referred to as Convolutional Bi-directional Long Short-Term Memory networks (CBLSTM)) has been built to address tool wear prediction tasks. Finally, Hinchi *et al.* [22] proposed a deep neural network for rolling elements bearing prognostic, by integrating a convolutional layer with LSTM layer (in a series fashion). However, to the best of our knowledge, only the above mentioned few attempts have been made towards development of hybrid solutions for RUL estimation. The paper addresses this gap.

Contributions: The paper proposes the HDNN framework for RUL estimation, as shown in Fig. 1. The proposed HDNN framework consists of two parallel paths (one LSTM and one CNN) followed by fully connected multilayer neural network, which fuses the output of each path to form the target RUL. The LSTM is used to extract temporal features while simultaneously the CNN is utilized to extract spatial features. It is worth mentioning that our proposed HDNN is the first hybrid deep neural network model for RUL estimation, that integrates two deep learning models to achieve this task. The proposed HDNN framework is tested and validated using the commercial modular aero-propulsion system simulation (C-MAPSS) data set by NASA [23]. Our comprehensive experiments and comparisons with over 5 recently proposed RUL estimation algorithms developed based on the same data-sets show that the proposed HDNN framework significantly outperforms all its counterparts especially in the complicated prognostic scenarios with increased number of operating conditions and fault modes.

The reminder of the paper is organized as follows: Section 2 develops the HDNN model. Experimental results and comparisons are provided in Section 3. Finally, Section 4 concludes the paper.

2. THE HDNN MODEL

In the proposed framework, a sliding window strategy is adopted to use the multi-variate temporal information, since a temporal sequence data provides more information in comparison to a multi-variate data point sampled at a single time step [24]. The input of the HDNN model is a two dimensional matrix containing r_{tw} (as the size of the sliding window) with r_f (as the number of the selected features). In the proposed HDNN framework, different values of r_{tw} have been considered for different sub-datasets of the C-MAPSS datasets. Moreover, the step size of sliding a window is chosen to be 1. The segmented multivariate time series matrix ($r_f \times r_{tw}$) is fed into the CNN path of the proposed model, while for the LSTM path, each column of the matrix ($r_f \times r_{tw}$) will be considered as an input to the LSTM at each time step. The structure of the three main components of the HDNN framework are as follows:

- ***CNN Path***: CNN is a multi-stage neural network which composed of some filter stages and one classification stage [25]. Three convolution layers are stacked in the CNN path of the HDNN model, for spatial features extraction. In the first two layers, 2-pairs of convolution layers and max pooling layers have been used, the convolution layers have the same configurations, i.e., 10 filters of size (10×1) are used, while the max pooling layers' filter size is (2×1). The third convolution layer is designed with one filter of size (3×1) to join the previous feature maps.
- ***LSTM Path***: The LSTMs are designed to overcome the vanishing and exploding gradient problem of the traditional Recurrent neural network (RNN) [26]. Three LSTM layers are stacked in the LSTM path of the HDNN model for temporal features extraction. Each of the first two layers is defined by 32 cell structure, while the third layer is based on 64 cell structure and repeating cells within the LSTM layer have the same structure and parameter values.
- ***Fusion Path***: Three fully connected layers for regression purpose to estimate the RUL. Each of the first two fusion layers has 100 neurons and uses "tanh" activation function. The third layer has 1 neuron and uses Rectified Linear Unit (ReLU) activation function. The input to the fusion layers is constructed as follows: The output of the CNN path (a 2-dimensional feature map) is flattened and then concatenated with the output features of the LSTM path. The resultant vector will be applied as the fusion input.
- ***Training***: The objective of the training process is to minimize the cost function by obtaining optimal parameters (weights and biases) [27]. The proposed approach aims to minimize the mean squared error (MSE) as it is the adopted cost function in this study, which is given by MSE $= \frac{1}{M_{tr}} \sum_{i=1}^{M_{tr}} h_i^2$, where M_{tr} is the total number of training data samples, and $h_i = \overline{RUL}_i - RUL_i$, i.e., the estimated RUL - true RUL with respect to the i_{th} data point. The mini batch gradient descent method [28] is used and the batch size is set equal to 512. In addition the Adam algorithm [29] is used for optimization.

Table 1. Data Set Details (Simulated From C-MAPSS) [32].

Dataset	C-MAPSS			
	FD001	**FD002**	**FD003**	**FD004**
Train Trajectories	100	260	100	249
Test Trajectories	100	259	100	248
Conditions	1	6	1	6
Fault Modes	1	1	2	2

3. EXPERIMENTS

In this section, we evaluate the performance of the proposed HDNN framework. First, the dataset descriptions is provided in Sub-section 3.1 followed by the details of the experimental setup. Finally, comparison results are shown and discussed in Sub-section 3.4.

3.1. NASA C-MAPSS Data Set

In this paper, the proposed approach (HDNN) is implemented and evaluated based on the degradation data sets of the turbofan engine provided in References [23] and [30]. The NASA C-MAPSS data set is a widely used benchmark data, generated using the NASA's propriety system level model-based simulation program software (named C-MAPSS). The C-MAPSS is a software for simulating the effects of faults and deterioration at different operating conditions in the main five rotating components (Fan, Low Pressure Compressor, High Pressure Compressor (HPC), High Pressure Turbine, and Low Pressure Turbine) found in a large commercial turbofan engine. The data set has divided into four sub-data sets (labeled from FD001 to FD004) with different number of operating conditions and fault modes. Each sub-data set is further divided into training and test subsets. The data sets outlined in Table 1. The data sets are arranged in an N-by-26 matrix, where N corresponds to the number of data points in each data set. Each row is a snapshot of data taken during a single operating time cycle, which includes 26 columns and each column represents a different variable. The 26 columns of data consist of two index values representing the engine number and the current operational cycle number, three operational settings that have a substantial effect on engine performance, as well as 21 sensor values, details of which can be found in [31]. Each trajectory within the data sets simulates the lifetime of an engine. While each engine is simulated with different initial conditions, the operational status of each engine is healthy in the early stage and begins to degrade as time progresses until a failure occurs. For each engine trajectory within the training sets, the last data entry corresponds to the moment the engine is declared unhealthy. While, trajectories in the test sets terminate at some time prior to failure and the target is to predict the number of cycles until the end of product lifetime for each engine commonly referred to as the RUL. The actual RUL value of the test trajectories for the C-MAPSS data set was made available to the public.

3.1.1. Operating Conditions and Fault Modes

The data points are classified into different distinct clusters using two factors, namely, the operating conditions and the associated fault

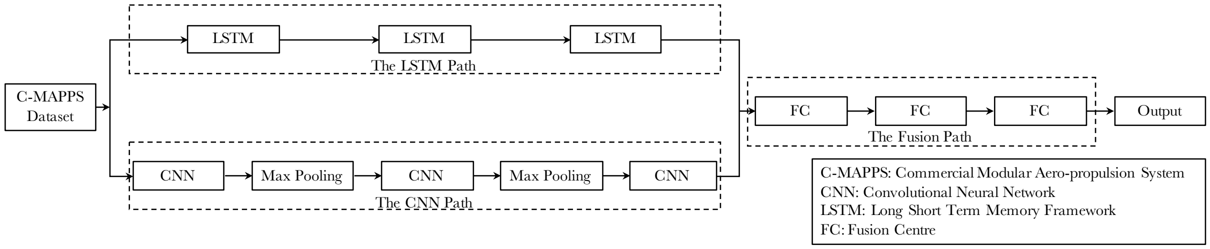


Fig. 1. The proposed hybrid deep neural network (HDNN) framework.

modes. As for the operating conditions, FD001 and FD003 are simulated at a single point (sea level), while FD002 and FD004 are simulated at six different operating conditions. For the fault modes, FD001 and FD002 are simulated with only HPC degradation, while FD003 and FD004 are simulated with both HPC and fan degradation resulting in more complex and challenging RUL predictions.

3.2. Data Normalization

A group of 14 sensor outcomes have been selected following Reference [33] as some sensor readings are not informative for RUL estimation, since they have almost constant outputs in the engine's life time. Min-max normalization has been used to enable the unbiased contribution from the output of each sensor, i.e.,

$$\bar{x}_i = \frac{2(x_i - \min x_i)}{\max x_i - \min x_i} - 1, \tag{1}$$

where x_i is the time sequence of i_{th} sensor measurements, and $\bar{x}_i$ is the normalized sensor data. This normalization will guarantee equal contribution from all features across all operating conditions [34]. The normalized data will be between [-1,1].

3.3. Evaluation Metrics

In this paper, two performance measures, i.e., scoring function, and Root Mean Square Error (RMSE), are used as briefly outlined below:

(1) *Scoring Function*: The scoring function used in this paper is the function that employed by the International Conference on Prognostic and Health Management (PHM08) Data Challenge and is given by

$$S = \sum_{i=1}^{n} s_i, \text{where} \quad s_i = \begin{cases} e^{-\frac{h_i}{13}} - 1 & for \quad h_i < 0 \\ e^{\frac{h_i}{10}} - 1 & for \quad h_i \geq 0, \end{cases}$$

where s is the computed score, n is the total number of testing data samples, and $h_i = \overline{RUL}_i - RUL_i$ (estimated RUL - true RUL, with respect to the i_{th} data point).

(2) *The RMSE*: It is commonly used as a performance measure since it gives equal weights for both early and late predictions. The formulation of the RMSE is given by RMSE $= \sqrt{\frac{1}{n}\sum_{i=1}^{n} h_i^2}$.

For this data-set a piece-wise linear RUL target function was proposed [34], which sets the maximum RUL to a constant value (based on the observations) and then the health of the system degrades linearly with usage.

3.4. Results

In this sub-section, we present various experimental results to evaluate the performance of the proposed HDNN framework for RUL estimation.

Table 2. The results of 30 time window size.

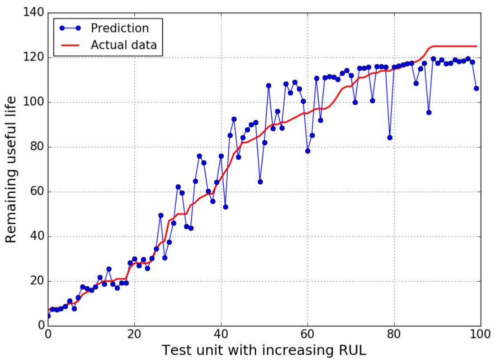
Metrics	FD001	FD002	FD003	FD004
Score	245	1282.42	287.72	1527.42
RMSE	13.017	15.24	12.22	18.156

3.4.1. The RUL Estimation Results

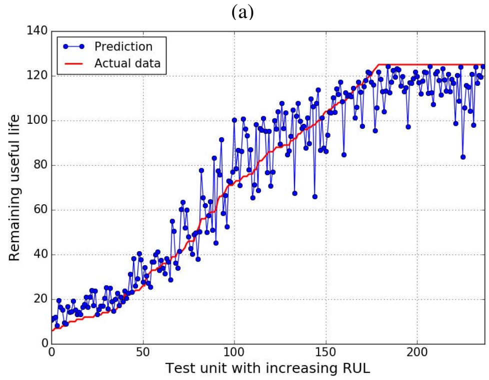
Figs. 2, and 3 present the RUL prediction results over two complex datasets (i.e., FD002 and FD004). in Figs 2(a) and (b) testing engines are sorted in an ascending order (from small to large). Figs. 2(a)-(d) show the prediction results associated with the last recorded data point for both datasets. It is noteworthy that the number of test cases in each dataset is different, where 256 and 248 engines in FD002 and FD004, respectively. It is observed that the predicted RUL values closely follow their ground truth. Two key point can be highlighted: (i) First, it can be noticed that engines with lower RUL are clearly with higher accuracy. This is particularly important as lower RUL translates to closeness of a potential failure and better accuracies are required to perform CBM actions at optimum times to avoid catastrophic failures, and; (ii) The results shown in Figs. 2(a) and (b) are significantly interesting as these are corresponding to the two most complex scenarios and typically existing solutions fail to provide reliable results for these two cases. Figs. 3(a) and (b) predicted RUL values for a sample unit of both dataset (selected in random). In par with our previous results, it is noticed that the proposed HDNN framework performs clearly well over both datasets, which are considered extremely complex scenarios and existing algorithms, typically, fail to provide precise predictions for these cases. More interestingly, the proposed HDNN framework manages to provide accurate RUL estimates values closely following their ground truth when the units are close to failure.

3.4.2. The Results with Different Time Window Size

Table 2 shows the results obtained from the proposed HDNN framework using 30 time window size for FD001 to FD004. Both the RMSE and the score values for FD002 and FD004 are substantially better than the values reported in the literature. To further show the efficiency of our proposed HDNN framework, we examine the effects of the window size on the results. As such, we used a smaller window size of 15 specially for FD002 and FD004 datasets, which have complex nature with more operating conditions in comparison to FD001 and FD003. It is observed that the results obtained from smaller time window size were again outstanding, i.e., for FD002 the HDNN achieves score of 1966.415 and RMSE of 17.411, which are considered exceptional. Similarly, for FD004 the score value obtained from a smaller window size is equal to 2549.64 with the RMSE value of 20.275. Comparison results with over 5 recent and state-of-the-art algorithms are presented next.



(a)



(b)

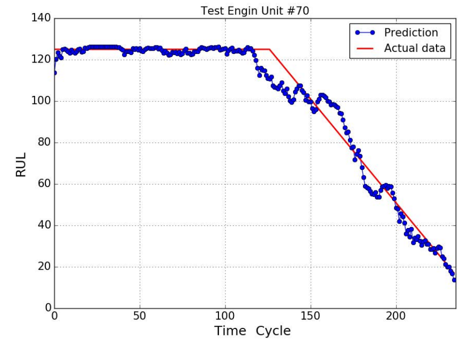
Fig. 2. The prediction for last recorded data point of different testing engine unites. (a) Prediction for the 256 testing engine unites in FD002. (b) Prediction for the 248 testing engine unites in FD004.

Table 3. Performance Comparison of 5 Methods with the proposed method (HDNN) based on C-MAPSS datasets.

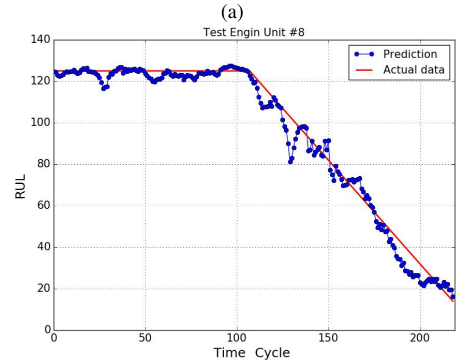
TW=30		HDNN	DCNN [35]	Rulclipper [36]	MODBNE [37]	FADCNN [38]	DLSTM [39]
FD001	Score	245	273.7	**216**	334.23	1286.7	338
	RMSE	13.017	**12.61**	13.27	15.04	18.45	16.14
FD002	Score	**1282.42**	10412	2796	5585.34	13570	4450
	RMSE	**15.24**	22.36	22.89	25.05	30.29	24.49
FD003	Score	287.72	**284.1**	317	421.91	1596.2	852
	RMSE	**12.22**	12.64	16	12.51	19.82	16.18
FD004	Score	**1527.42**	12466	3132	6557.62	7886.4	5550
	RMSE	**18.156**	23.31	24.33	28.66	29.16	28.17

3.4.3. Comparison with Existing Methods

To evaluate and exactly point the placement of the reported results compared to the existing literature we present comprehensive comparisons with existing state-of-the art RUL estimation solutions that used the C-MAPSS datasets. Table 3 illustrates the results reported by the most successful approaches for RUL prediction in the literature and are compared with the proposed HDNN framework. Amongst these works, it is worth mentioning that our proposed HDNN is the first hybrid deep neural network model for RUL estimation, that integrates two deep learning models to achieve this task. It can be clearly seen from Table 3 that the proposed HDNN model conducted almost the best performance and achieved the best outcomes among the existing approaches. The exceptional improvement in the outcomes distinguishes our model as it is the pioneer that achieved such outcomes, specifically for FD002 and FD004. The achieved improvements in the outcomes, as compared with the best results available in the literature (DCNN) [35], exceeds 20% in terms of the RMSE value, and 80% in terms of the score value for FD002, while 13% improvement is achieved in terms of the RMSE value, and 80% in terms of the score value for FD004. In sum, the proposed HDNN framework is capable of achieving exceptional and unprecedented outcomes as shown in Table 2. Aside from the great results achieved for the FD002 and FD004, the results accomplished from utilizing the FD001 and FD003 are also outstanding and are better than most of the existing approaches except the DCNN method, where slightly outperforms the HDNN, i.e., about 3% in terms of the RMSE value for FD001, and 1% in terms of the score value for FD003. We would like to point out that the HDNN results are still better about 11% in terms of the score value for FD001, and 3% in terms of the RMSE for FD003.



(a)



(b)

Fig. 3. Different examples of life time RUL prediction for a sample engine unit of each dataset. (a) The testing engine Unit 70 in FD002. (b) The testing engine Unit 8 in FD004.

4. CONCLUSION

In this paper, we have proposed and demonstrated a new deep learning approach referred to as Hybrid Deep Neural Network Model (HDNN) for RUL estimation from multi-variate sensor signals. The proposed HDNN framework is a hybrid architecture that integrates a deep LSTM and a deep CNN coupled via fusion and fully connected layers, to achieve an exceptional outcomes. The proposed method has been tested on the NASA's C-MAPSS dataset that simulates the degrading health of a commercial aero engine. Comparisons with several state-of-the-art methodologies have been conducted and the results demonstrate the outstanding performance of the proposed HDNN, specifically on complex datasets consisting of several operating conditions and fault modes.

5. REFERENCES

[1] Z. Huang, Z. Xu, X. Ke, W. Wang, and Y. Sun, "Remaining Useful Life Prediction for an Adaptive Skew-Wiener Process Mode," *Mech. Syst. Signal Process.*, vol. 87, pp. 294-306, 2017.

[2] C. Chen, B. Zhang, G. Vachtsevanos, and M. Orchard, "Machine Condition Prediction based on Adaptive Neuro–fuzzy and High-order Particle Filtering," *IEEE Trans. Ind. Electron.*, vol. 58, no. 9, pp. 4353-4364, 2011.

[3] Z. Tian, L. Wong, and N. Safaei,, "A Neural Network Approach for Remaining Useful Life Prediction Utilizing both Failure and Suspension Histories," *Mech. Syst. Signal Process.*, vol. 24, no. 5, pp. 1542–1555, 2010.

[4] C. Hu, B. D. Youn, P. Wang, and J. T. Yoon, "Ensemble of Data-driven Prognostic Algorithms for Robust Prediction of Remaining Useful Life" *Rel. Eng. Syst. Safety*, vol. 103, pp. 120-135, 2012.

[5] K. Liu, N. Z. Gebraeel and J. Shi, "A Data-Level Fusion Model for Developing Composite Health Indices for Degradation Modeling and Prognostic Analysis," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 3, pp. 652-664, 2013.

[6] P. Lim, C. K. Goh, K. C. Tan, and P. Dutta, "Estimation of Remaining Useful Life based on Switching Kalman Filter Neural Network Ensemble," *Proc. Int. Conf. Prognostics Health Manage. (PHM)*, pp. 2-9, 2014.

[7] M. Kan, A. Tan, and J. Mathew, "A Review on Prognostic Techniques for Non-stationary and Non-linear Rotating Systems," *Mech. Syst. Signal Process.*, PP. 1-20, 2015.

[8] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. Zhao, "Time Series Classification using Multi-channels Deep Convolutional Neural Networks," *proc. 15th Int. Conf. Web-Age Inf. Manag.*, pp. 298-310, 2014.

[9] Z. Chen, C. Li, and R. Sanchez, "Gearbox Fault Identification and Classification with Convolutional Neural Networks," *Shock and Vibration*, 2015.

[10] D. Lee, V. Siu, R. Cruz, and C. Yetman, " Convolutional Neural Net and Bearing Fault Analysis," *in Proc. Int. Conf. Data Min.*, pp. 194-200, 2016.

[11] J. Wang, J. Zhuang, L. Duan, and W. Cheng, "A Multi-scale Convolution Neural Network for Featureless Fault Diagnosis," *International Symposium of Flexible Automation (ISFA)*, pp. 1-6, 2016.

[12] G. S. Babu, P. Zhao, and X. Li, "Deep Convolutional Neural Network based Regression Approach for Estimation of Remaining Useful Life," *International Conference on Database Systems for Advanced Applications*, pp. 214-228, 2016.

[13] W. Sun, R. Zhao, R. Yan, S. Shao, and X. Chen, "Convolutional Discriminative Feature Learning for Induction Motor Fault Diagnosis," *IEEE Trans. Ind. Inf.*, vol. 13, no. 3, pp. 1350-1359, 2017.

[14] X. Li, Q. Ding, and J. Sun "Remaining Useful Life Estimation in Prognostics using Deep Convolution Neural Networks," *Reliability Engineering and System Safety*, vol. 172, pp. 1-11, 2018.

[15] L. Ren, Y. Sun, H. Wang, and L. Zhang "Prediction of Bearing Remaining Useful Life With Deep Convolution Neural Network," *IEEE Special Section on Cyber-Physical-Social Computing and Networking*, vol. 6, pp. 13041-13049, 2018.

[16] P. Malhotra, V. TV, A Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "Multi-Sensor Prognostics using an Unsupervised Health Index based on LSTM Encoder-Decoder," *ACM SIGKDD Workshop on Machine Learning for Prognostics and Health Management*, 2016.

[17] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long Short-Term Memory Network for Remaining Useful Life Estimation," *IEEE International Conference on Prognostics and Health Management (ICPHM)*, pp. 88-95, 2017.

[18] Y. Wu, M. Wu, S. Dong, L. Lin, and Y. Liu, "Remaining Useful Life Estimation of Engineered Systems using Vanilla LSTM Neural Networks," *Neurocomputing*, vol. 275, pp. 167-179, 2018.

[19] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery Health Prognostics: A Systematic Review from Data Acquisition to RUL Prediction," *Mech. Syst. & Signal Process.*, vol. 104, pp. 799-834, 2018.

[20] Y. Zhang, R. Xiong, H. He and M. G. Pecht,, "Long Short-Term Memory Recurrent Neural Network for Remaining Useful Life Prediction of Lithium-Ion Batteries," *IEEE Trans. Veh. Technol*, vol. 67, no. 7, pp. 5695-5705, 2018.

[21] R. Zhao, R. Yan, J. Wang, and K. Mao, "Learning to Monitor Machine Health with Convolutional Bi-Directional LSTM Networks," *Sensors*, vol. 17. no. 2, 2017.

[22] A. Zakariae, and Z. Hinchi, "Rolling Element Bearing Remaining Useful Life Estimation based on a Convolutional Long-short-term memory Network," *Procedia Computer Science*, vol. 127, pp. 123-132, 2018.

[23] A. Saxena and K. Goebel, "C-MAPSS Data Set," *NASA Ames Prognostics Data Repository*, 2008.

[24] L. Xiang, D. Qian, S. Jian-Qiao, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering and System Safety*, vol. 172, pp. 1-11, 2018.

[25] W. Lu, B. Liang, Y. Cheng, D. Meng, J. Yang, and T. Zhang "Deep model based domain adaptation for fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 3, pp. 2296 - 2305, 2017.

[26] Y. Zhang, R. Xiong, H. He and M. G. Pecht,, "Long Short-Term Memory Recurrent Neural Network for Remaining Useful Life Prediction of Lithium-Ion Batteries," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 5695-5705, 2018.

[27] Iliya Valchanov, "Machine Learning: An Overview," "https://www.datascience.com/blog/machine-learning-overview," 2018.

[28] Jason Brownlee, "A Gentle Introduction to Mini-Batch Gradient Descent and How to Configure Batch Size," https://machinelearningmastery.com/gentle-introduction-mini-batch-gradient-descent-configure-batch-size/, 2017.

[29] D. Kingma, and J. Ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv* 14126980, 2014.

[30] A. Saxena and K. Goebel, (2008). "PHM08 Challenge Data Set," *NASA Ames Research Center, Moffett Field, CA, USA. [Online]. Available: http://ti.arc.nasa.gov/project/prognostic-data-repository*, 2008.

[31] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage Propagation Modeling for Aircraft Engine Run-to-failure Simulation," *Proc. IEEE Int. Conf. Prognostics Health Manage.*, pp. 1-9, 2008.

[32] L. Pin, C. Goh, T. Kay, P. Dutta, "Multimodal Degradation Prognostics Based on Switching Kalman Filter Ensemble," *IEEE Trans. Neural Networks & Learning Systems*, vol. 28, no. 1, pp. 136-148, 2017.

[33] C. Zhang, P. Lim, A. K. Qin and K. C. Tan, "Multiobjective Deep Belief Networks Ensemble for Remaining Useful Life Estimation in Prognostics," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2306-2318, 2017.

[34] L. Peel, "Data Driven Prognostics using a Kalman Filter Ensemble of Neural Network Models," *Proc. Int. Conf. Prognostics Health Manage. (PHM)*, pp. 1-6, 2008.

[35] X. Li, Q. Ding, and J. Sun "Remaining Useful Life Estimation in Prognostics using Deep Convolution Neural Networks," *Reliability Engineering and System Safety*, vol. 172, pp. 1-11, 2018.

[36] E. Ramasso, "Investigating Computational Geometry for Failure Prognostics," *Int J Prog- nostics Health Manage*, vol. 5, no. 1, 2014.

[37] C. Zhang, P. Lim, A. K. Qin and K. C. Tan, "Multiobjective Deep Belief Networks Ensemble for Remaining Useful Life Estimation in Prognostics," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2306-2318, 2017.

[38] G. S. Babu, P. Zhao, and X. Li, "Deep Convolutional Neural Network based Regression Approach for Estimation of Remaining Useful Life," *International Conference on Database Systems for Advanced Applications*, pp. 214-228, 2016.

[39] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long Short-Term Memory Network for Remaining Useful Life Estimation," *IEEE International Conference on Prognostics and Health Management (ICPHM)*, pp. 88-95, 2017.