

RESEARCH ARTICLE



Dynamic Neural Network Architecture Design for Predicting Remaining Useful Life of Dynamic Processes

Silvio Simani^{1,*} , Yat Ping Lam², Saverio Farsoni¹ and Paolo Castaldi³

¹Department of Engineering, University of Ferrara, Italy

²Department of Engineering, The Chinese University of Hong Kong (Shenzhen), China

³Department of Electrical, Electronic and Information Engineering, University of Bologna, Italy

Abstract: The prediction of remaining useful life is critical in predictive health management. This is done to reduce the expenses associated with operation and maintenance by avoiding errors and failures in dynamic processes. Recently, the abilities of feature classification and automated extraction of neural networks in its convolutional forms have shown fascinating performance when used for estimating the remaining useful life of dynamic processes using deep learning structures. This was accomplished by putting these talents to the task of predicting how long the procedures would be beneficial. Existing network topologies, on the other hand, virtually entirely extract features at a single scale while neglecting important information at other sizes. Meanwhile, because of the architecture of a single network path, the comprehensiveness of the features discovered by these tools is limited. To address these concerns, the authors propose a network structure based on a feature fusion strategy on a parallel multiscale architecture. This structure is then utilized to compute the remaining useful life. This prototype is divided into two sections: the first is a multiscale feature extraction module designed to extract local information features, and the second is a causal convolution module designed to extract global information features by combining multi-layer causal convolution with average pooling. The multiscale feature extraction module is intended for the extraction of local information features, while the causal convolution module is intended for the extraction of global information features. Finally, the two distinct paths are joined to create a fully integrated layer. The simulations and results show that this method has the potential to improve the efficiency and accuracy of estimating the remaining useful life index. Furthermore, the advantages of the established strategy are shown by comparing the results obtained with those produced by applying cutting-edge techniques on a well-known data-set depicting a simulated turbofan engine.

Keywords: health monitoring, remaining useful life, deep learning, regression model, aero-engine data set

1. Introduction

The increasing demand and use of industrial equipment is a direct result of the rising productivity in today's modern, industrialised society. The estimate of the Remaining Useful Life (RUL) indicator is a vital problem in predictive health management (PHM), i.e. the reconstruction of the time left before defects or failures occur and therefore a promising area of technology. To prevent catastrophic failures in dynamic processes, the technical staff responsible for the plant's upkeep must be provided with an effective RUL pertaining to the processes or components under surveillance. This is essential for carrying out operations and maintenance effectively, leading to successful maintenance activities and choices [1]. To minimise the likelihood of failure due to a problem, it is essential to improve and establish a proper operating and maintenance schedule. Because of this, we may anticipate a reduction in expenses.

However, thanks to the quick and substantial deployment of computer infrastructures and facilities, deep learning using deep neural networks has begun to become an intriguing and successful research subject, even in the prediction and estimation frameworks. The reason for this is because they are better than previous methods in capturing hierarchical connections in deep structures [2]. Therefore, the goal of this study is to provide an accurate RUL reconstruction by means of a deep learning technique, which is applied here in the form of convolutional neural networks (CNNs).

Degenerate model schemes, data-driven techniques and fusion methods are the usual foundations for RUL estimates. Key to RUL reconstruction are model-based approaches, which often make use of models generated from physical laws characterising the operating circumstances of the dynamic process [3].

To complicate matters further, rapid technological advancement has rendered modern components and their tools exceedingly difficult to analyse in depth; in particular, it is challenging to define an effective failure model that can be used in practice and real-time conditions [4].

*Corresponding author: Silvio Simani, Department of Engineering, University of Ferrara, Italy. Email: smsnlv@unife.it

Data-driven schemes, on the other hand, do not take into account the physical laws that govern the mechanisms and the effective working conditions of the entire process; instead, they provide static or dynamic relationships between historical data sequences acquired from the process sensors, and the RUL estimations are derived using data mining methodologies [1]. It is possible to use data-driven strategies to construct RUL estimates using parametric and nonparametric methods under these circumstances if enough sequences of operational data can be collected.

To offer an estimate of the RUL metric, hybrid techniques may use degradation models based on combination processes, which are included in data-driven schemes [5]. Unfortunately, efficient degradation models with multiple processes may be expensive to develop and implement, which places a premium on hybrid techniques. Due to its usefulness in real-world scenarios and cutting-edge contexts, data-driven methods have recently gained considerable traction.

The major goal of these data-driven schemes is to adaptively construct approximation descriptions for degradation models, the processes of which make use of measurements and the knowledge of the RUL categorisation. Here, RUL estimation has been accomplished using a variety of techniques, including shallow auto-regressive prototypes [6], Wiener processes [7] and Bayesian approaches [8]. RUL prediction becomes more difficult because these schemes need attention to feature engineering and feature quality. The authors in Xia et al. [9], Zhang et al. [10], and Chen et al. [11] all point to the growing interest in RUL prediction using CNNs as examples of deep learning's general-purpose tools.

CNNs have been demonstrated to be capable of fitting complicated signals in recent deep learning research. These CNNs can retain local spatial correlation while being scale-, shift-, and distortion-invariant. By employing data collected for condition monitoring, these structures provide a clear roadmap for extracting fault characteristics. However, even the most recent methods are not without their flaws. While basic CNN may work for short-term data, they struggle to extract information effectively and reliably from longer-term sequences. Truth be told, there is a great deal of superfluous data in sensor readings that will cloud the model's ability to make informed judgments. The drawback is that the information at other scales is lost because of the fixed filter size used to collect the characteristics. The multiscale deep convolutional neural network (MS-DCNN) is superior to other networks in feature extraction because its design includes filters of varying size at each layer.

This paper proposes a new parallel structured network (PSN) for RUL prediction of industrial equipment, which uses causal convolution and multiscale extraction architecture as two submodules for the same input processed simultaneously in two paths to improve prediction accuracy, address the high complexity and account for the time series nature of RUL data characteristics. To begin, we build two modules that can process the input simultaneously. Module 1's convolution layer incorporates causal attributes, and it feeds a mean pooling layer to get the global features. The module 2 tool is the multiscale feature extraction framework. To further enhance local feature selection, a channel attention module has been included. Then, to produce a feature fusion of global and local information, the knowledge gained by the two modules is coupled. A fully connected layer then takes the combined output and uses it to do data regression analysis and

provide an RUL estimate. The findings obtained verify that the suggested method can reliably forecast the RUL for each component. This is also a significant improvement over the most recent findings on the same data set, so it is important to keep that in mind.

New research has shown that multiscale learning strategies have promising applications in many fields, including but not limited to scene and image resolution management, autonomous vehicle design and medical diagnostics [12, 13]. By developing numerous CNN prototypes with different input sizes all at once in the same setting, multiscale architectures can learn and extract features from a variety of scales. Using this method, characteristics are processed by several models before being combined into fully linked modules [14].

U-Nets, which stand for CNNs with multiscale feature fusion, were the focus of study by Ronneberger et al. [15] in the target identification and segmentation tasks. The experiments that employed this solution came out well. Target features were also abstracted layer by layer in Takacs et al. [16], and feature information from many scales was combined to enhance the detection impact. With a multiscale approach in the width direction, convolutional kernels of varying sizes are employed to represent perceptual fields of varying dimensions in Szegedy et al. [17]. This improved visual representation is the result of the stitching process, which allows for the fusing of elements at various scales.

While first developed for computer vision applications, multiscale feature fusion networks are seeing increasing use in research applications focused on the establishment of approaches for the estimate of the RUL performance indicator. For instance, the CNN presented by Li et al. [18] uses layers of filters of varying sizes to choose multiscale features from converted frequency sequences into temporal spectrograms. Their conclusions were more precise than those of simpler single-scale structures. Contrarily, in this study by Wang et al. [19], convolutional structures with varying dilation frequencies were combined to produce a larger convolutional module that improved prediction accuracy when trained with information from several sensor types. In order to pinpoint the specifics of deterioration, the publication Wang et al. [20] provided a method for comparing sequences collected by various sensors. The use of several time sequences at varying frequencies also allowed for the development of a multiple scale learning structure that could automatically construct representations.

As a result of these considerations, this work makes use of a multiple scale feature learning approach to create a technique that can reliably estimate the RUL indication. Training the network to anticipate the necessary indices is made easier by being able to discern the granularity of the features resulting from the usage of several scales.

As a result, this paper's goals are outlined and summarised below considering the relevant and more recent state-of-the-art tactics.

- 1) In this work, the benefits of using causal convolution in the processing of time series are investigated. The improvements of multiscale extraction architecture are also thoroughly incorporated. Additionally, a PSN is recommended to offer the estimate of the RUL indication when it is applied to aero-engine data; as a result, the traditional network structure is altered.
- 2) The created PSN is able to extract features with multiscale properties by using filters with varied dimensions; this is favourable to the network's ability to learn in a more comprehensive manner. After the multiscale feature extraction

module has been completed, the channel attention module is implemented to improve the performance of the multiscale features, further emphasise the information that is vital to the channel and boost the potential of the features to learn.

- 3) In order to increase the accuracy of the estimate of the RUL index, it is possible to make use of the causal convolution in order to extract the correlation from data sequences that display long-term dependencies.

As a last point of discussion, it is crucial to highlight that the need to examine and limit the complexity of an architecture may be prompted by several events, the majority of the time in order to reduce the amount of processing that is necessary. However, complexity cannot be arbitrarily reduced since it is the only model that produced excellent results after several cycles of training and testing. This issue is actively being researched, and the outcomes of that study are being included into this work. This paper examines several alternative solutions to the same problem that CNNs confront when implemented in a variety of frameworks. CNNs, for example, are impeded by the fact that they have an excessive number of trainable parameters, which reduces their computational efficiency. Consequently, the complexity reduction problem is investigated at the end of the task, and numerous conclusions are drawn from that investigation.

In conclusion, the structure of the remaining portions of the work is as follows. In Section 2, an analysis of comparable contributions from recent research on the state of the art is presented. These contributions use frameworks for the learning of multiscale features and the estimate of the RUL indicator. The architecture that uses the deep learning principle and is exploited for the purpose of deriving strategies that are able to estimate the RUL index is described in full in the section that can be found in Section 3. The sequences that are utilised for the experimental validation of the solutions presented in Section 5 are detailed in detail in Section 4. These sequences were taken from a well-known benchmark of an aero-propulsion system data set [21], which may be found in Section 5. The findings that were obtained help to illustrate the effectiveness and advantages of the suggested PSN, which is also contrasted with several other methods. Last but not least, the study is brought to a close in Section 6, which provides a brief summary of the most significant accomplishments and identifies certain unresolved issues that need more research and analysis.

2. RUL Estimation with Deep Learning Architectures

The architecture relying on deep learning structures exploited for providing the estimation of the RUL indicator and applied to the turbofan engine data considered in this work is described in Section 2.1. First, a general description of the network's component architecture is given, and then, the components are described in detail in Section 2.2.

2.1. RUL prediction methods

The estimation of the RUL indicator has been investigated for decades in the related literature. In the recent state of the art, the significant advancement in CNNs and deep learning methodologies, many artificial intelligence schemes based on these principles have been developed for providing the reconstruction of the RUL indicator. Therefore, some strategies that exploit these tools are addressed in this section.

As already remarked, the increasing interest in these artificial intelligence tools in connection with deep learning and CNNs has led to good results in predicting remaining life expectancy studies. In recent years, CNN, networks implementing long-short-term memory (LSTM) modules and units with recursive gates (GRU)s can be considered when the application is oriented to the estimation of the RUL indicator, and good results need to be achieved. By analysing similar strategies, the paper by Li et al. [6] developed a new strategy for the prediction of the lifetime of a dynamic process via a deep CNN (DCNN) based on constructing samples with time-varying windows. The samples were prepared by time windows to better achieve feature extraction, and experiments on the aero-propulsion system data set [21], which are exploited by this paper, highlighted that the reconstruction of the RUL index was more accurate than other mainstream research approaches. On the other hand, the work by Zhu et al. [22] considered an approach using CNN tools for the estimation of the RUL indicator for multiple scale bearings. This method showed that the global and local information can be kept synchronised by using convolutional kernels of different sizes compared with the conventional CNN. Kim and Sohn [23] proposed a CNN structure using multiple task learning algorithms to highlight the correlation between the RUL reconstruction and the healthy status of the monitored plant. The paper by Li et al. [24] proposed a novel directed acyclic graph (DAG) structure used as prediction prototype that integrates the LSTM module into a CNN. This prototype was developed for predictive maintenance applications to turbofan engine systems. By replacing the conventional linear function with a degradation mechanism and segmented remaining life function, the operational status of turbofan engines can be effectively graded. The work by Al-Dulaimi et al. [25] addressed a neural network with an hybrid structure for the estimation of the RUL indicator. To this end, LSTM modules were considered again to derive possible temporal correlations among the features, while CNN was proposed to obtain features in the spatial domain. This approach showed interesting prognostic characteristics when applied to complex processes. The authors in Zhu et al. [26] described an end-to-end approach to the RUL index estimation relying on feature fusion, which combines attention mechanisms, CNNs, and bidirectional GRUs (BGRU)s to achieve multiple feature fusions. Moreover, the work by Li et al. [27] developed a solution using MS-DCNN models for aero-engine RUL prediction that were exploited to extract simple static and dynamic links between health monitoring sequences and the RUL indicator. The manuscript by Liu et al. [28] used a BGRU model to derive correlations in long-term sequences from filtered input sequences; moreover, a CNN was introduced to extract the features with local characteristics from the output data generated by the BGRU module.

Compared with the problem considered in this paper, LSTM still has some shortcomings when the elaboration of extended time sequences with many samples. This issue derives from the limited information stored in memory cells. Therefore, the work by Lea et al. [29] addressed a temporal convolutional network (TCN) which, having more sensory fields, achieved higher prediction performance over long histories by introducing the null causal convolution. Therefore, the network structure in temporal convolution can be exploited for the problem of the estimation of the RUL indicator. On the other hand, given that TCNs exhibit significantly longer memory, the work by Chen et al. [30] investigated the first application of TCN to the RUL estimation. Immediately after, the paper by Chen et al. [31] developed a genetic algorithm optimised RUL prediction model based on

TCN, which improved the estimation accuracy of the system, thus providing an effective solution for the condition monitoring of turbofan engine systems. On the other hand, Zeng et al. [32] suggested a new neural network including deep attention residual (DARNN) prototypes for machine RUL prediction. Moreover, time convolution was used as the basic building module of the developed deep attention residual (DAR) block, which allowed to obtain good prediction results.

To further enhance the performance of CNN, the most common and practical way is the principle of attention. This characteristic feature originated from the analysis of human visual system: the visual perception mechanism implies that the human eye focuses on what it wants to highlight on in the visual field and ignores other things consciously, and it was first applied to machine translation tasks [33]. In recent years, also the attention mechanism (AM) has been applied to the field of the estimation of the RUL indicator. As an example, Song et al. [34] developed a novel AM scheme to be applied to the reconstruction of the RUL index. In this way, the problem experimented with existing methods that do not directly understand the links between several time sequences of data has been overcome, thus improving the effectiveness in the estimation of the RUL indicator. The work by Muneer et al. [35] proposed a DCNN including the AM, which was exploited to provide the estimation of the RUL indicator for turbofan plants with good results. On the other hand, Fan et al. [36] addressed a new methodology with end-to-end features for solving the RUL prediction task. In this approach, an efficient and powerful AM was specifically designed to increase the estimation capabilities, which was relying on the analysis of the signal characteristics for the RUL prediction task. Tseng and Tran [37] included the AM block into the LSTM structure, thus combining the information from the previously hidden layers and the input modules into the actual state to effectively discriminate which new feature needs to be added to the storage module.

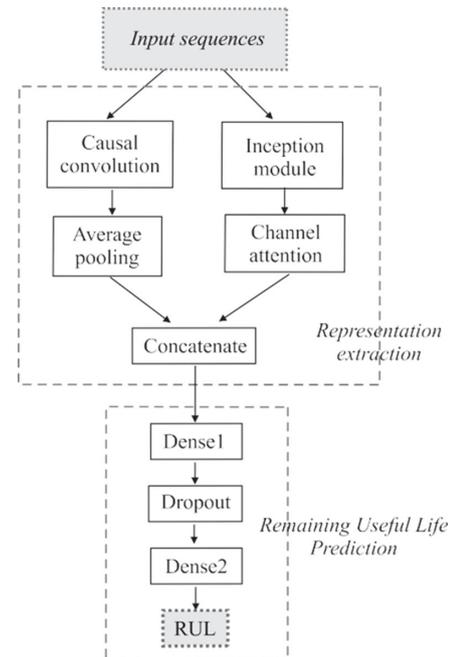
Based on the above literature review, the effective capabilities of CNNs to extract features allow to obtain good results when they are applied to fault monitoring by using input and output sensor data sequences. The implementation of the AM in the CNN structures further improves the effectiveness of the RUL estimation. Null causal convolution also shows a better prediction performance in longer time series. However, most current networks perform only feature extraction of single network paths and do not well fuse the advantages of multiple networks. Considering the need to both grasp the relationship between long time series data globally and capture the local degradation feature characteristics, this work suggests exploiting parallel multiple scale CNN with fusion mechanisms for the reconstruction of the RUL indicator.

2.2. Deep learning architectures

The architecture relying on the strategies suggested in this manuscript, i.e. the methodology using PSN prototypes, is shown in Figure 1.

As shown in Figure 1, the data sequences feed the PSN architecture as parallel inputs. In this way, these inputs enter a causal convolution module and an inception block. On one hand, causal convolutions are a kind of convolution that is utilised for temporal data. They ensure that the model does not violate the order in which we model the data. In particular, the model's forecast at one timestep cannot depend on any of the succeeding timesteps. A masked convolution is the visual equivalent of a

Figure 1
Diagram of the parallel structured network (PSN) architecture



causal convolution. A masked convolution may be performed by first producing a mask tensor and then doing an element-wise multiplication of this mask with the convolution kernel before applying it. This is the same as a causal convolution. When dealing with one-dimensional data, such as audio, this is considerably easier to implement simply shifting the output of a standard convolution forward or backwards by a few timesteps. On the other hand, Inception modules are used in CNNs to enable more effective computation and deeper networks by lowering the network's dimensionality via the use of stacked 1×1 convolutions. This has been completed. The modules were created to give a solution to the problem of computational cost as well as overfitting and other issues. In a nutshell, the solution is to arrange various kernel filter sizes within the CNN to act on the same level rather than stacking them sequentially. This will fix the issue.

According to Figure 1, the information flow continues through the average pooling and the channel attention modules. The method known as "Average Pooling" is a kind of pooling that builds a down-sampled feature map by computing the average value for each patch of a feature map and then using that average value to construct the down-sampled feature map. In most situations, it comes after a layer that implements the convolutional approach. It adds a minor bit of translation invariance, which implies that translating the image by a small amount has little effect on the values of most pooled outputs. This is because the modest amount of translation invariance that it offers is additive. Max Pooling may extract features that are more conspicuous, such as edges, while this approach can extract features that are extracted with more smoothness. A "Channel Attention Module" is a component of a CNN that oversees channel-based attention. We may create a channel attention map by using the inter-channel connectivity of attributes. Because each channel of a feature map is viewed as a unique feature detector, channel attention is focused on "what" is significant in the context of the image being processed. We

reduce the spatial dimension of the input feature map in order to improve the efficiency of channel attention calculation.

Finally, Figure 1 highlights that the two parallel flows processed according to these modules are concatenated, thus resulting in a sharing of information. Then, dense and dropout blocks follow the signal processing to achieve the RUL estimation. CNNs can have one kind of layer, known as the dense or completely connected layer. Because each output neuron considers all input neurons, this layer is known as being completely connected. As a result, the number of parameters that the network must learn is on the order of magnitude that corresponds to the product of the number of input neurons and the number of hidden neurons. On the other hand, there are many different types of layers, each of which performs a unique set of calculations and takes a unique collection of input. This research considers convolutional layers as well as dropout layers, both of which are useful not just in the context of image data but also in a broad range of other types of (organised) data.

Therefore, in PSN, the time series achieved via different measurements from the monitored dynamic process are employed as inputs to extract and identify multiple channel information. Moreover, these inputs are learned in the causal convolution submodule and the multiscale feature extraction submodule, and then, the learning results of the two modules are fused. Finally, the fused results are fed to the structure including the fully connected module that is used to provide the RUL estimation. This PSN prototype is analysed with more details in the next sections.

In this manuscript, in the causal convolution module, after the operation of three layers of causal convolution, the averaging pooling operation is performed to reduce the model size, decrease the number of variables, and reconstruct the overall details regarding the extracted features. This multiple scale feature extraction module exploits different convolution kernel operations to extract local information at different scales, while combining channel attention to strengthen the weight of important local information, while suppressing redundant information. Therefore, this paper suggests to employ feature fusion concatenation (FFC) in order to extract and fuse the feature information of these two modules, which is commonly used in both fully CNN (U-Net) and dense CNN (DenseNet) architectures, as shown in Ronneberger et al. [15] and Huang et al. [38]. These methodologies can effectively integrate the feature information, while reducing the dimension of the CNN at the same time. However, the width of the CNN is also improved, so that this architecture can be trained by using more features, thus leading to an improvement of the prediction effectiveness and its accuracy.

This work assumes that input sequences are organised as two-dimensional variables, where the first dimension represents the number of features, while the second one indicates the length of the time sequences. However, considering that the mechanical features collected for solving this prediction problem are acquired from different sensors, the links among features that are spatially adjacent and present in the time sequences can be neglected. In this way, although the input sequences and the related maps of features present two-dimensional characteristics, in practice, the filter implementing the convolution operation and included in the proposed architecture has only one dimension.

3. Parallel Structured Neural Network Prototypes

The structure of the considered PSN proposed in this work is analysed in this section and exploited for the reconstruction of the RUL indicator.

3.1. Causal convolution module

Temporal convolution achieves excellent results on RUL prediction, thanks to the inclusion of a causal convolution module (submodule 1) with an expansion rate. In this paper, causal convolution (CC) is used to enhance the capability of the developed deep learning model to extract proper representations. For sequential problems, usually RNN or LSTM are exploited. In fact, the major advantage of CNN with respect to RNN consists of its capabilities to share weights and the availability of a convolution module with local awareness features. When the weight parameters are shared, the number of variables in the network that must be trained can be reduced. Moreover, local perception module can provide a smoother representation of the information regarding the local features characterising the input time sequences managed by the actual convolutional module. Thus, causal convolution can not only understand the correlation time with long-term characteristics among the input historical sequences but also execute computations in parallel mode as the case of CNN architectures.

In the following, causal and dilated convolution operations are considered.

On one hand, unlike traditional convolutional operations, the output of the module performing the convolution task with causal behaviour when considered at the instant t is only convolved with the signals at the instant t and previous samples. This represents the causal convolutional block of the CNN, which thus enables a sequence of samples generated as moving average of its lagged inputs. In this way, the proposed architecture does not lose any information provided to the network from the oldest to the newest samples of the sequence, as highlighted in Zeng et al. [32].

This task can be specifically described in mathematical form as in Equation (1):

$$y(t+1) = F(u(1), u(2), \dots, u(t-1), u(t)) \quad (1)$$

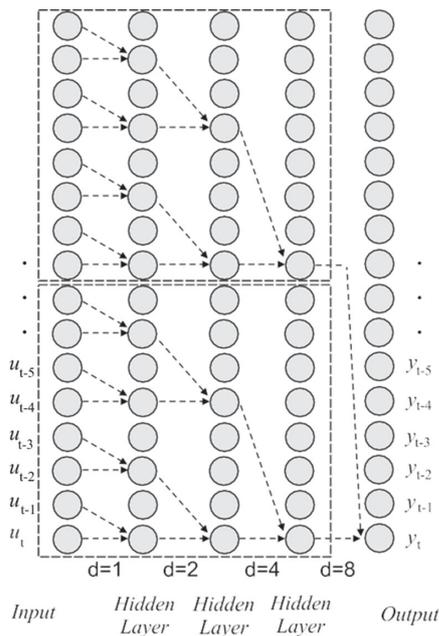
where $u(t)$ is a one-dimensional vector containing n dimensional features, and y_{t+1} is the value of the variable to be represented in the current next time period. $F(\cdot)$ is the function that establishes the mathematical relationship between u and y . However, one limitation of causal convolution is that the coverage of historical information is not large, and each output value can only be associated with a small portion of the input information when the layers of the network are deepened. Therefore, an extended convolution operation is suggested in this paper for processing the signals in a causal way, to solve this problem.

Secondly, a different convolution operation with a dilated mode enables to include different sampling times from several time sequences, where the number of acquired samples is defined by the parameter d . This block and its dynamics are shown in Figure 2.

In particular, the higher level presents a delay $d = 1$. This represents the case where each sample is acquired from the considered input; on the other hand, an intermediate level can present a delay parameter $d = 2$, thus implying that two samples are feeding the considered layer. In general, as highlighted in Figure 2, the lower layer includes the larger number of delays d that are computed there. In this way, it is evident that this convolution operation with dilated effects leads to moving window with a size that increases exponentially depending on the considered number of levels (layers).

This architecture of CNN leads to generate extended perceptual field by including only a limited number of layers. Moreover, this convolution processing with causal features that includes a dilated

Figure 2
Causal convolution that uses dilated rate strategy



sampling frequency scheme is thus sketched in Figure 2, with reference also to the relation of Equation (1).

3.2. Average pooling module

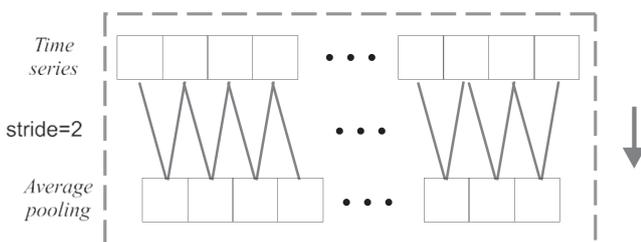
The principle of the pooling block consists of sampling and mapping the considered feature representations after the convolution operation. The essence of this process is to compress and downscale the feature maps, which simplifies the model parameters while filtering out the important feature information and avoiding the redundancy of useless feature information, which not only speeds up the model computation but also improves the robustness of the model. Averaging pooling performs the selection of regional averages that tends to preserve the characteristics of the overall data. The process of averaging pooling for 1D convolution is shown in Figure 3.

3.3. Multiple scale feature extraction module

This section is focused on the submodule 2 of the architecture of Figure 1 and in particular the multiscale extraction network.

In CNN, the design of its structure and, in particular, the selection of the dimension of its convolutional kernel represent a

Figure 3
Example of average pooling of one-dimensional convolution



key point. This depends on the fact that these kernels implementing the convolutional features with varying dimensions allow the extraction of the information by considering variable time scales. Specifically, it is better to use large convolutional kernels for information distributed on a global scale and small convolutional kernels for data distributed on a local scale.

In this study, attention is made to the inception module in the model structure of inception-v3. The inception module embeds multiscale information and aggregates features on multiple different sensory fields to obtain performance gains. The architecture of inception-v3 [17, 39] represents a CNN specifically designed for allowing image processing, their analysis and feature detection. It was built as block for GoogLeNet. In its third version, it appears as module of the Google’s Inception CNN [40], originally introduced during the challenge focusing on image recognition, i.e. ImageNet, considered in Russakovsky et al. [41].

The structure of inception-v3 was proposed to develop deeper neural architectures networks, while maintaining a reduced number of variables. For example, it can easily reach about 25 million variables, which can become more than 60 million in the model of AlexNet [17]. It can be considered as a collection and classification of visual features, thus enhancing the recognition of objects, when considering the computer vision domain [39]. Moreover, the prototype relying in inception-v3 has been exploited and applied to a huge number of different examples and often employed as pre-trained architecture by ImageNet. One example of its common application can be found in life sciences, since it helps to investigate research topics regarding leukaemia, as described in Poojary and Pai [42].

The inception module has four paths of convolutional and pooling layers with different hyperparameters to extract different information, which is quite equal to a module with four streams. In this way, it is used to extract features and working in parallel with convolutional blocks implementing windows of different dimensions, as well as layers with maximum pooling modules. In particular, the architecture exploits those modules with a one-dimensional convolutional block, to limit the complexity of the structure. Moreover, also the channel dimensions are reduced, thus presenting a small number of parameters and low computational complexity. The scheme of the inception module considered in this paper is sketched in Figure 3.

3.4. Attention module block

Neural structures consider different streams of several feature representations that are usually exploited for interpreting different features. In particular, the attention module performs the object selection process in each channel, which can selectively tune its weights with an adaptive strategy, thus allowing to focus on the specific channel, depending on the considered task.

The feature maps for each channel have varying levels of accuracy depending on the status of the monitored machinery. Some inputs may represent fundamental aspects that are extremely important for monitoring the machine conditions, while other channels can be useless due to the low level of signal-to-noise ratio.

Therefore, the attention module applied to channels is added to the multiple scale CNN to further highlight the important input information. A representative of the channel attention model is Squeeze-and-Excitation Net (SENet), as addressed in Wang et al. [43]. In this way, SENet is chosen as a part of the CNN architecture proposed in this paper, as sketched in Figure 1.

4. Experimental Setup

The paper considers the time series from the set known as C-MAPSS, which represents a benchmark for testing solutions regarding the estimation of the RUL indicator. As reported in Table 1, these sequences consist of 4 subsets, labelled as “FD001”, “FD002”, “FD003”, and “FD004”. Each group of data contains training and test data. The sequences indicated as “FD002” and “FD004” are more challenging than “FD001” and “FD003”. Moreover, they include more training and testing examples, while involving six different working situations, especially “FD004” or 2 failure modes. Therefore, this subset implies the most complex task. These data sets will be shortly referred to “set i ”, with the index i varying from 1 to 4, respectively. More details on these time sequences are available in Ramasso [44], Thakkar and Chaoui [45], and Berghout et al. [21].

The training sequences are characterised by multivariate signals acquired from the nominal engine. After one working period, a first fault commences, and then, the number of malfunctions increases. At the end of the time series, each multivariate set is concluded with a completely faulty machine. Differently from the training data, the test sequence terminates before the machine is faulty.

To perform the feature selection, in the above 24-dimensional feature data, the sets 1 and 3 consist of single working situations. In this way, the time sequences that contain three fixed working situations can generate high correlations among the features extracted with degraded characteristics. Moreover, since the data acquired from the sensors 1, 5, 6, 10, 16, 18, and 19 present anomalous conditions, only the data from 14 sensors can be exploited as inputs to the model.

Before constructing this model, the original sequences need to be filtered to have normalised features. This pre-processing of the data is usually performed to enhance the convergence speed of the training and estimation algorithms. In particular, this study proposes to exploit a min-max pre-filtering for normalisation purpose. Therefore, each sample of the sequences $u^{i,j}$ assumes a normal value in the range between $[-1, 1]$. The relation to compute this transformation has the form of Equation (2):

$$u_n^{i,j} = 2 \frac{u^{i,j} - u_m^j}{u_M^j - u_m^j} - 1 \tag{2}$$

where $u^{i,j}$ represents the i -th signal from the j -th sensing device, $u_n^{i,j}$ is the corresponding filtered value of $u^{i,j}$. The terms u_M^j and u_m^j stand for the maximum and minimum samples taken from the original signals acquired by the j -th sensing device, respectively.

For RUL prediction, a piecewise affine degradation prototype is required. In order to fulfil this assumption, the component or equipment is in normal condition for a longer period of time

Table 1
Time sequence details

Details	Set 1	Set 2	Set 3	Set 4
Training cases #	100	260	100	249
Testing cases #	100	259	100	248
Max. fault-free cases	362	378	525	543
Min. fault-free cases	128	128	145	128
Measured signals	21	21	21	21
Working situations	1	6	1	6
Fault cases	1	1	2	2

throughout its life cycle. Therefore, it is considered a segmented linear behaviour for the degradation dynamics that is applied to the C-MAPSS data from the turbofan engines, as described in Ramasso [44]. In most cases, once the RUL index assumes values around 116, the health state of the equipment starts to change at this point, so the maximum RUL is limited to about 116 in the experiments of this paper. The considered behaviour of the monitored machine assumes that it usually performs in the early conditions, while its life does not vary. If a fault affects the monitored process, its lifetime starts to decrease in a linear way. Finally, the machine is completely failed. These assumptions represent reasonable and practical issues when a general degradation model is considered for describing the behaviour of plant equipment or component.

As further data pre-processing, by aggregating the data in a time window, this makes the data smoother, while minimising the effect of noise on the features used in the model. In multivariate time series-based problems (such as the estimation of the RUL index), more details can be usually acquired from time sequences than from multivariate samples obtained at single sampling frequency. Thus, the processing of data sequences with causal features provides better predictive performance. The most suitable time window size is taken for each subset in the experiments performed in this paper. These time window sizes for the sets 1, 2, 3 and 4 are equal to 36, 20, 36 and 19, respectively.

Moreover, in the experiments considered in this paper, the effectiveness of the developed approach is evaluated by using a suitable performance function, which provides a proper score value, as described in the following Wang et al. [46]. On the other hand, the normalised root mean square error (NRMSE) index is employed to analyse the normalised average estimation error, i.e. the normalised difference between the predicted and the actual values of the considered signals. The NRMSE is computed as shown in Equation (3):

$$NRMSE = \sqrt{\frac{\sum_{i=1}^N (RUL_i - \hat{RUL}_i)^2}{\sum_{i=1}^N RUL_i^2}} \tag{3}$$

where the index i refers to the instant, for a number of data N .

Note that the NRMSE index weights the different samples with the same attention. However, the scoring indicator assumes higher values when the estimated value of the RUL indicator is smaller than its actual one. This fact is due to the property that ‘over-predicting’ the RUL indicator may imply important mechanical failures or even catastrophic incidents. The score index has the form of Equation (4):

$$\text{score} = \begin{cases} \sum_{i=1}^N \left(e^{-\frac{R\hat{U}L_i - RUL_i}{13}} - 1 \right), & R\hat{U}L_i < RUL_i \\ \sum_{i=1}^N \left(e^{-\frac{R\hat{U}L_i - RUL_i}{10}} - 1 \right), & R\hat{U}L_i \geq RUL_i \end{cases} \tag{4}$$

where RUL_i and $R\hat{U}L_i$ denote the actual and the estimated RUL values at the i th time index, respectively.

With reference to the hyperparameter configuration considered in this paper, the NRMSE performance index is exploited for evaluating the fitness of the estimated model, tuning the model parameters and improving the reconstruction capabilities of the estimated model. Finally, the parameter combination setting, i.e. the so-called model structure, corresponding to the model that leads to optimal estimations is shown in Table 2. The values of

Table 2
List of hyperparameter configurations

Hyperparameter	Value
CC kernel size k	5
Dilated rate d	2
Window size	512
Dropout size	0.5
Step size	0.001 ÷ 0.0001
1 st FC neuron #	100
2 nd FC neuron #	64

the neurons included in the Fully Connected (FC) modules are also summarised.

5. Achievements and Comparisons

This section summarises the achievements that were obtained to illustrate the effectiveness and advantages of the suggested PSN, which is also contrasted with several other methods. The sequences that are utilised for the experimental validation of the solutions presented in Section 5 were detailed in detail in Section 4. These sequences were borrowed from the well-known benchmark of an aero-propulsion system data set [21]. The robustness and sensitivity characteristics of the proposed schemes are also investigated.

5.1. Nominal performance analysis

In Figures 4, 5, 6 and 7, the normalised errors for the estimations of the RUL indicator are shown, when the data sets 1, 2, 3 and 4 from the test engine unit are considered, with reference to the final recorded time sequences. These test modules are labelled by starting from the small ones and then taking the large units, thus improving the analysis and the estimation results. In particular, Figures 4, 5, 6, 7 and 8 report the values of the normalised prediction error (NPE) indicator, as defined by the relation of Equation (5):

$$NPE_i = \frac{RUL_i - \hat{RUL}_i}{RUL_i} \quad (5)$$

where the index i refers to the i th sample of the sequence.

Figure 4
Diagram of the inception module structure

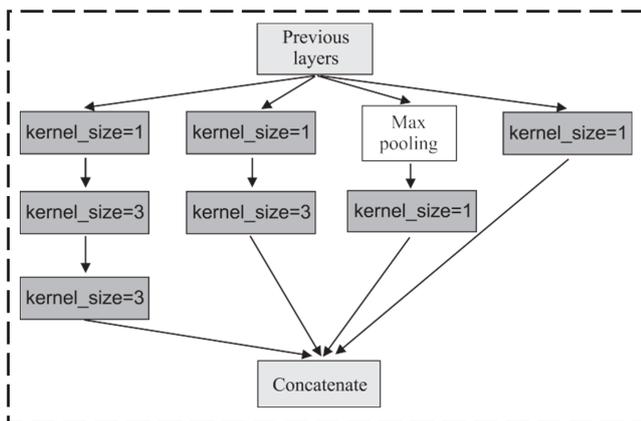


Figure 5
NPE_i values for the testing data set 1

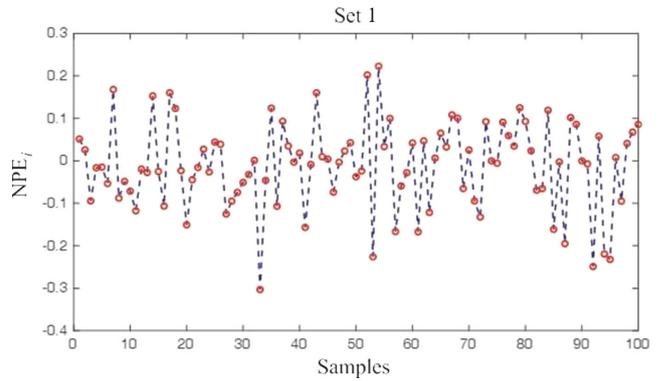


Figure 6
NPE_i values for the testing data set 2

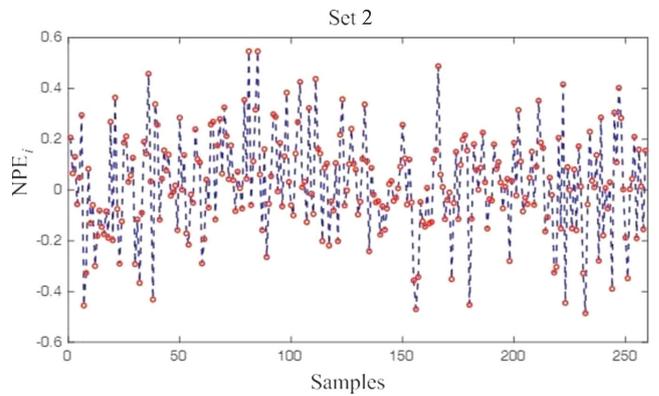
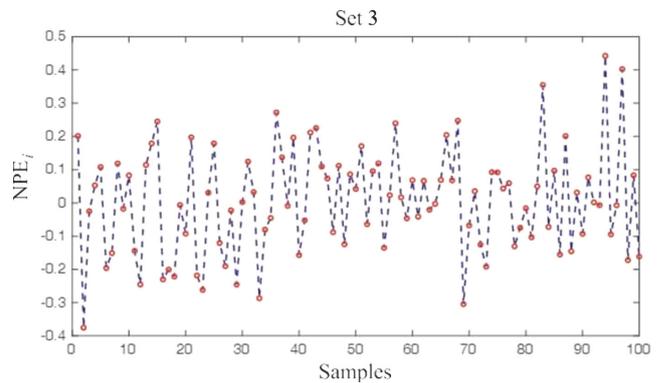
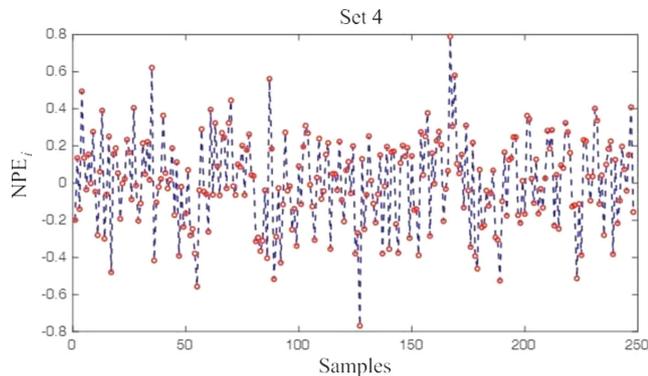


Figure 7
NPE_i values for the testing data set 3



It is worth noting that the prediction errors of the RUL indicator provided by the developed scheme and that are related to the NRMSE of Equation (3) are very small. In particular, the prediction effectiveness results to be more accurate when the RUL indicator assumes small values. This result is motivated by the fact that if the engine device does not work properly, the suggested approach

Figure 8
NPE_i values for the testing data set 4



enables the extraction of the fault features, thus enhancing the prediction efficacy.

In addition, the estimation effectiveness is higher for the data sets 1 and 3, with respect to the one achievable with the sets 2 and 4. This behaviour can be explained by observing that the engines exhibit very complicated working situations described by the sequences from the sets 2 and 3. Therefore, the estimation of their RUL indicators is more difficult. In general, the proposed strategy seems to provide very good results when the RUL estimation is related to mechanical systems.

The C-MAPSS data sets considered in this work represent a very common test bed used in research studies that investigate the RUL estimation problem; in fact, as highlighted at the beginning of this paper, a significant number of contributions has analysed this topic recently. Moreover, in order to highlight the effectiveness of the scheme developed in this study, RUL prediction experiments are performed for the data sets 1, 2, 3 and 4. Moreover, they are also analysed by taking into account the achievements addressed in some recent paper. To this end, Tables 2 and 3 summarise these works.

Therefore, Tables 3 and 4 highlight how this work, by means of the solution addressed here, is able to obtain optimal results in terms of NRMSE metrics compared to other strategies. With reference to the data set 3, all metrics computed for the solution developed here reach the optimum. In addition, the strategy developed by this study leads to significantly better performance than other methodologies when the most complicated and difficult sequences are

Table 3
Performance analysis for the data sets 1 and 2

Method	Set 1		Set 2	
	NRMSE	Score	NRMSE	Score
DCNN [6]	0.0261	256	0.0326	7865
AGCNN [28]	0.0241	267	0.0498	2345
CNN-BiLSTM [34]	0.0231	214	0.0501	2104
MT-CNN [23]	0.0284	218	0.0497	2345
DARNN [32]	0.0265	216	0.0392	1012
TaFCN [36]	0.0343	301	0.0234	2012
GA-TCN [31]	0.0218	298	0.0432	1675
End-to-end [26]	0.0231	287	0.0476	1768
PSN (this paper)	0.0117	198	0.0148	998

Table 4
Performance analysis for the data sets 3 and 4

Method	Set 3		Set 4	
	NRMSE	Score	NRMSE	Score
DCNN [6]	0.0364	301	0.2336	9876
AGCNN [28]	0.0339	345	0.0234	3567
CNN-BiLSTM [34]	0.0496	301	0.0567	1978
MT-CNN [23]	0.0299	401	0.0567	2134
DARNN [32]	0.0345	365	0.0672	2678
TaFCN [36]	0.0378	267	0.0456	3897
GA-TCN [31]	0.0401	389	0.0345	1956
End-to-end [26]	0.0298	302	0.0378	2867
PSN (this paper)	0.0187	199	0.0169	1674

represented by the data sets 2 and set 4. Thus, the proposed method is promising for the RUL prediction tasks.

With reference to the methods reported in Tables 3 and 4, Li et al. [6] considered a data-driven strategy for health management relying on DCNN, while Liu et al. [28] developed another CNN that with adaptive Gabor (AGCNN) structure and receptive fields. On the other hand, Song et al. [34] developed a solution aimed at resolving the limits shown by common state-of-the-art approaches unable to extract relations between historical series, which decrease the efficacy of RUL reconstruction. To this end, a CNN structure with an attention module and a bidirectional LSTM (CNN-BiLSTM) block was developed. On the other hand, Kim and Sohn [23] considered a CNN model with multi-task (MT-CNN) features implementing a training algorithm able to extract the relationships between the RUL prediction and the diagnostics task. Moreover, Zeng et al. [32] discovered that the prediction performance may extremely depend on the working conditions of the process under investigation. These further limitations were managed by developing a new DARNN.

On the other hand, by considering the features of the signals used for RUL estimation, the authors in Fan et al. [36] proposed a new Fully CNN architecture including a trend attention module that enhanced the estimation capabilities. However, it can be difficult to fix the parameters of the designed architecture and improve the efficacy of the RUL estimation. Therefore, to solve this issue, the RUL estimation strategy was enhanced by a genetic algorithm, which integrated a TCN, as described in Chen et al. [31] and denoted as GA-TCN. Finally, different from the previous approaches, Zhu et al. [26] designed a new framework integrating a feature fusion method and an end-to-end scheme oriented to RUL estimation, thus combining spatial and temporal operations and the extraction of raw characteristics (end-to-end).

Note that the causal operation module and the multiple scale block for the identification of the relevant features represent key aspects of the overall architecture. To highlight how this affects the efficacy of the proposed solution, an ablation analysis is carried out in the following.

To highlight the benefits of causal convolution and multiscale feature extraction modules, four different architectures have been also proposed to estimate the RUL indicator. These models are defined as NN-1, NN-2 and NN-3. These three prototypes have the following structures:

- 1) NN-1 does not include the causal convolution and the multiple scale training algorithm, and uses a single-path conventional convolutional network;

- 2) NN-2 implements just the causal operation module;
- 3) NN-3 contains just the multiscale feature extraction module.

In addition, the other parameters are kept constant. To demonstrate the efficacy of the solution developed in this work, the behaviour of each part on the four different subsets is shown. The achievements applied to the real data are listed in Tables 5, 6, and 7.

Tables 5 and 6 show that the fused method (index “sum”) summarised in Table 7 performs better in most subsets compared to the two separate modules, and although the two submodules perform better with the data from the set 2, it differs less, and the fused model demonstrates significant advantages in both the NRMSE and the score when comparing the sum of the evaluated metrics. Therefore, these results are sufficient to demonstrate the efficacy and superiority of the PSN architecture developed in this work with respect to the considered and different strategies available in the recent and related literature.

5.2. Robustness and sensitivity features

The evaluation of the performances of the considered architecture is based again on the computation of the NRMSE index of Equation (3).

A proper Monte Carlo analysis has been performed to compute these indices and to test the robustness of the considered scheme. Indeed, the Monte Carlo tool is useful at this stage, as the efficacy of the proposed solutions depends on both the model approximation capabilities and the measurements errors. The same tool was exploited by the authors for testing the reliability features of fault diagnosis schemes, as described in Patton et al. [47, 48].

Table 5
Ablation results for the data sets 1 and 2

Model	Set 1		Set 2	
	NRMSE	Score	NRMSE	Score
NN-1	0.02139	267	0.0252	2134
NN-2	0.02337	245	0.0244	1987
NN-3	0.02156	213	0.0214	1135
PSN	0.0117	169	0.0136	667

Table 6
Ablation results for the data sets 3 and 4

Model	Set 3		Set 4	
	NRMSE	Score	NRMSE	Score
NN-1	12.33	501	0.0385	2768
NN-2	12.07	342	0.0276	2567
NN-3	12.04	302	0.0257	2167
PSN	10.07	155	0.0121	1577

Table 7
Comparison results of the index sum

Architecture	Sum	
	NRMSE	Score
NN-1	0.0727	5321
NN-2	0.0719	4123
NN-3	0.0659	3456
PSN	0.0520	3135

Table 8
Sensitivity analysis results for the data sets 2, 3 and 4 in terms of NRMSE index

Method	Set 2	Set 3	Set 4
DCNN [6]	0.0822	0.0864	0.07931
AGCNN [28]	0.0943	0.0733	0.0599
CNN-BiLSTM [34]	0.0601	0.0696	0.0618
MT-CNN [23]	0.0597	0.0451	0.0765
DARNN [32]	0.0592	0.0618	0.0782
TaFCN [36]	0.0716	0.0601	0.0797
GA-TCN [31]	0.0677	0.0701	0.0962
End-to-end [26]	0.0360	0.0323	0.0398
PSN (this paper)	0.0284	0.0207	0.0268

In particular, a set of 1000 Monte Carlo runs has been performed, during which realistic uncertainties have been considered by modelling some meaningful variables of the time series from the C-MAPSS data sets as Gaussian stochastic processes around their nominal values and with standard deviations corresponding to the realistic minimal and maximal error values computed by the relation of Equation (2).

Therefore, also in this case, Table 8 shows that, by means of the strategy proposed here, allows to obtain very good results in terms of NRMSE and score metrics compared to the other methodologies. Again, with reference to the data set 3, all metrics computed for the structure proposed here lead to the best results, also in the presence of uncertainty. Moreover, the solution developed by this paper leads to significantly better performance than other schemes available in the related literature, when disturbance affects the sequences represented by the data sets 2 and set 4. Thus, the achieved results demonstrate that the proposed method is promising for the RUL prediction tasks under uncertain and disturbance effects.

6. Conclusion

The most important aspect of predictive condition monitoring is determining the remaining usable life. This kind of monitoring may help to cut operating and maintenance costs while also lowering the risk of major incidents happening. The parallel multiscale feature fusion network for industrial equipment considered in this work incorporates the benefits of various network structures, considers both global and local feature information, selects two different network structures as two paths for the input data, and finally fuses the results, which has the potential to significantly improve the remaining useful life prediction accuracy. The goal of this effort was to advance the area of parallel multiscale feature fusion networks for industrial equipment. Significant improvements were gained when the proposed methods were compared to other benchmark approaches and applied to a well-established data set. Furthermore, when this comparison was performed, the supplied solutions produced the best outcomes. This was done to establish whether or not the responses offered represented an improvement. However, when real-world production processes are considered, the number of pieces of equipment might be limited. Furthermore, unscheduled equipment downtime is an incident that is unlikely to occur. As a result, there will be a restricted number of possible sequences that may be retrieved for diagnostic purposes. As a result, building machine learning models or techniques of model pre-training that are suited for small sample data sets will be another important research path to pursue when considering component lifetime estimates. This

is due to the fact that these approaches are best suited for data sets with a small number of observations.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

Data available on request from the corresponding author upon reasonable request.

References

- [1] Lei, Y., Li, N., Guo, L., Li, N., Yan, T., & Lin, J. (2018). Machinery health prognostics: A systematic review from data acquisition to RUL prediction. *Mechanical Systems and Signal Processing*, 104, 799–834. <https://doi.org/10.1016/j.ymssp.2017.11.016>.
- [2] Ma, M., Sun, C., & Chen, X. (2017). Discriminative deep belief networks with ant colony optimization for health status assessment of machine. *IEEE Transactions on Instrumentation and Measurement*, 66(12), 3115–3125. <https://doi.org/10.1109/TIM.2017.2735661>.
- [3] Pan, Z., Meng, Z., Chen, Z., Gao, W., & Shi, Y. (2020). A two-stage method based on extreme learning machine for predicting the remaining useful life of rolling-element bearings. *Mechanical Systems and Signal Processing*, 144, 106899. <https://doi.org/10.1016/j.ymssp.2020.106899>.
- [4] Mao, W., He, J., & Zuo, M. J. (2020). Predicting remaining useful life of rolling bearings based on deep feature representation and transfer learning. *IEEE Transactions on Instrumentation and Measurement*, 69(4), 1594–1608. <https://doi.org/10.1109/TIM.2019.2917735>.
- [5] Wang, B., Lei, Y., Li, N., & Li, N. (2020). A hybrid prognostics approach for estimating remaining useful life of rolling element bearings. *IEEE Transactions on Reliability*, 69(1), 401–412. <https://doi.org/10.1109/TR.2018.2882682>.
- [6] Li, X., Ding, Q., & Sun, J. Q. (2018). Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliability Engineering & System Safety*, 172, 1–11. <https://doi.org/10.1016/j.res.2017.11.021>.
- [7] Yu, W., Shao, Y., Xu, J., & Mechefske, C. (2022). An adaptive and generalized wiener process model with a recursive filtering algorithm for remaining useful life estimation. *Reliability Engineering & System Safety*, 217, 108099. <https://doi.org/10.1016/j.res.2021.108099>.
- [8] Duan, F., & Wang, G. (2022). Bayesian analysis for the transformed exponential dispersion process with random effects. *Reliability Engineering & System Safety*, 217, 108104. <https://doi.org/10.1016/j.res.2021.108104>.
- [9] Xia, M., Li, T., Shu, T., Wan, J., de Silva, C. W., & Wang, Z. (2019). A two-stage approach for the remaining useful life prediction of bearings using deep neural networks. *IEEE Transactions on Industrial Informatics*, 15(6), 3703–3711. <https://doi.org/10.1109/TII.2018.2868687>.
- [10] Zhang, G., Liang, W., She, B., & Tian, F. (2021). Rotating machinery remaining useful life prediction scheme using deep-learning-based health indicator and a new RVM. *Shock and Vibration*, 2021(1), 8815241. <https://doi.org/10.1155/2021/8815241>.
- [11] Chen, Y., Zhang, D., & Zhang, W. (2022). MSWR-LRCN: A new deep learning approach to remaining useful life estimation of bearings. *Control Engineering Practice*, 118, 104969. <https://doi.org/10.1016/j.conengprac.2021.104969>.
- [12] Zhang, R., Chen, J., Feng, L., Li, S., Yang, W., & Guo, D. (2022). A refined pyramid scene parsing network for polarimetric SAR image semantic segmentation in agricultural areas. *IEEE Geoscience and Remote Sensing Letters*, 19, 1–5. <https://doi.org/10.1109/LGRS.2021.3086117>.
- [13] Zhou, W., Lin, X., Lei, J., Yu, L., & Hwang, J. N. (2022). MFFNet: Multiscale feature fusion and enhancement network for RGB-thermal urban road scene parsing. *IEEE Transactions on Multimedia*, 24, 2526–2538. <https://doi.org/10.1109/TMM.2021.3086618>.
- [14] Chen, L. C., Zhu, Y., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision*, 801–818.
- [15] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015: 18th International Conference*, 234–241. https://doi.org/10.1007/978-3-319-24574-4_28
- [16] Takacs, B., Vincze, Z., Fassold, H., Karakottas, A., Zioulis, N., Zarpalas, D., & Daras, P. (2019). Hyper 360 - Towards a unified tool set supporting next generation VR film and TV productions. *Journal of Software Engineering and Applications*, 12(5), 127–148. <https://doi.org/10.4236/jsea.2019.125009>.
- [17] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., . . . , & Rabinovich, A. (2015). Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition*, 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>.
- [18] Li, X., Zhang, W., & Ding, Q. (2019). Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction. *Reliability Engineering & System Safety*, 182, 208–218. <https://doi.org/10.1016/j.res.2018.11.011>.
- [19] Wang, R., Shi, R., Hu, X., & Shen, C. (2021). Remaining useful life prediction of rolling bearings based on multiscale convolutional neural network with integrated dilated convolution blocks. *Shock and Vibration*, 2021(1), 6616861. <https://doi.org/10.1155/2021/6616861>.
- [20] Wang, B., Lei, Y., Li, N., & Wang, W. (2021). Multiscale convolutional attention network for predicting remaining useful life of machinery. *IEEE Transactions on Industrial Electronics*, 68(8), 7496–7504. <https://doi.org/10.1109/TIE.2020.3003649>.
- [21] Berghout, T., Mouss, M. D., Mouss, L. H., & Benbouzid, M. (2023). Prognnet: A transferable deep network for aircraft engine damage propagation prognosis under real flight conditions. *Aerospace*, 10(1), 10. <https://doi.org/10.3390/aerospace10010010>.
- [22] Zhu, J., Chen, N., & Peng, W. (2019). Estimation of bearing remaining useful life based on multiscale convolutional neural network. *IEEE Transactions on Industrial Electronics*, 66(4), 3208–3216. <https://doi.org/10.1109/TIE.2018.2844856>.
- [23] Kim, T. S., & Sohn, S. Y. (2021). Multitask learning for health condition identification and remaining useful life prediction: Deep convolutional neural network approach. *Journal of Intelligent Manufacturing*, 32(8), 2169–2179. <https://doi.org/10.1007/s10845-020-01630-w>.
- [24] Li, J., Li, X., & He, D. (2019). A directed acyclic graph network combined with CNN and LSTM for remaining useful life prediction. *IEEE Access*, 7, 75464–75475. <https://doi.org/10.1109/ACCESS.2019.2919566>.

- [25] Al-Dulaimi, A., Zabihi, S., Asif, A., & Mohammadi, A. (2019). A multimodal and hybrid deep neural network model for remaining useful life estimation. *Computers in Industry*, 108, 186–196. <https://doi.org/10.1016/j.compind.2019.02.004>.
- [26] Zhu, Q., Xiong, Q., Yang, Z., & Yu, Y. (2023). A novel feature-fusion-based end-to-end approach for remaining useful life prediction. *Journal of Intelligent Manufacturing*, 34(8), 3495–3505. <https://doi.org/10.1007/s10845-022-02015-x>.
- [27] Li, H., Zhao, W., Zhang, Y., & Zio, E. (2020). Remaining useful life prediction using multi-scale deep convolutional neural network. *Applied Soft Computing*, 89, 106113. <https://doi.org/10.1016/j.asoc.2020.106113>.
- [28] Liu, H., Liu, Z., Jia, W., & Lin, X. (2021). Remaining useful life prediction using a novel feature-attention-based end-to-end approach. *IEEE Transactions on Industrial Informatics*, 17(2), 1197–1207. <https://doi.org/10.1109/TII.2020.2983760>.
- [29] Lea, C., Flynn, M. D., Vidal, R., Reiter, A., & Hager, G. D. (2017). Temporal convolutional networks for action segmentation and detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition*, 1003–1012. <https://doi.org/10.1109/CVPR.2017.113>.
- [30] Chen, J., Chen, D., & Liu, G. (2021). Using temporal convolution network for remaining useful lifetime prediction. *Engineering Reports*, 3(3), e12305. <https://doi.org/10.1002/eng2.12305>.
- [31] Chen, Z., Chen, B., & Chen, X. (2022). Remaining useful life prediction of turbofan engine based on temporal convolutional networks optimized by genetic algorithm. *Journal of Physics: Conference Series*, 2181, 012001. <https://doi.org/10.1088/1742-6596/2181/1/012001>.
- [32] Zeng, F., Li, Y., Jiang, Y., & Song, G. (2021). A deep attention residual neural network-based remaining useful life prediction of machinery. *Measurement*, 181, 109642. <https://doi.org/10.1016/j.measurement.2021.109642>.
- [33] Feng, X., Feng, Z., Zhao, W., Qin, B., & Liu, T. (2020). Enhanced neural machine translation by joint decoding with word and POS-tagging sequences. *Mobile Networks & Applications* 25(5), 1722–1728. <https://doi.org/10.1007/s11036-020-01582-8>.
- [34] Song, J. W., Park, Y. I., Hong, J. J., Kim, S. G., & Kang, S. J. (2021). Attention-based bidirectional LSTM-CNN model for remaining useful life estimation. In *2021 IEEE International Symposium on Circuits and Systems*, 1–5. <https://doi.org/10.1109/ISCAS51556.2021.9401572>.
- [35] Muneer, A., Taib, S. M., Fati, S. M., & Alhussian, H. (2021). Deep-learning based prognosis approach for remaining useful life prediction of turbofan engine. *Symmetry*, 13(10), 1861. <https://doi.org/10.3390/sym13101861>.
- [36] Fan, L., Chai, Y., & Chen, X. (2022). Trend attention fully convolutional network for remaining useful life estimation. *Reliability Engineering & System Safety*, 225, 108590. <https://doi.org/10.1016/j.res.2022.108590>.
- [37] Tseng, S. H., & Tran, K. D. (2024). Predicting maintenance through an attention long short-term memory projected model. *Journal of Intelligent Manufacturing*, 35, 807–824. <https://doi.org/10.1007/s10845-023-02077-5>.
- [38] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition*, 2261–2269. <https://doi.org/10.1109/CVPR.2017.243>.
- [39] Tang, J. (2018). *Intelligent mobile projects with TensorFlow: Build 10+ artificial intelligence apps using TensorFlow mobile and lite for iOS, Android, and Raspberry Pi*. UK: Packt Publishing.
- [40] Khosravi, P., Kazemi, E., Imielinski, M., Elemento, O., & Hajirasouliha, I. (2018). Deep convolutional neural networks enable discrimination of heterogeneous digital pathology images. *EBioMedicine*, 27, 317–328. <https://doi.org/10.1016/j.ebiom.2017.12.026>.
- [41] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., . . . , & Bernstein, M. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3), 211–252. <https://doi.org/10.1007/s11263-015-0816-y>.
- [42] Poojary, R., & Pai, A. (2019). Comparative study of model optimization techniques in fine-tuned CNN models. In *2019 International Conference on Electrical and Computing Technologies and Applications*, 1–4. <https://doi.org/10.1109/ICECTA48151.2019.8959681>.
- [43] Wang, T., Huan, J., & Zhu, M. (2019). Instance-based deep transfer learning. In *2019 IEEE Winter Conference on Applications of Computer Vision*, 367–375. <https://doi.org/10.1109/WACV.2019.00045>.
- [44] Ramasso, E. (2014). Investigating computational geometry for failure prognostics in presence of imprecise health indicator: Results and comparisons on C-MAPSS datasets. In *PHM Society European Conference*, 2(1), 1–13. <https://doi.org/10.36001/phme.2014.v2i1.1460>.
- [45] Thakkar, U., & Chaoui, H. (2022). Remaining useful life prediction of an aircraft turbofan engine using deep layer recurrent neural networks. *Actuators*, 11(3), 67. <https://doi.org/10.3390/act11030067>.
- [46] Wang, Q., Zheng, S., Farahat, A., Serita, S., & Gupta, C. (2019). Remaining useful life estimation using functional data analysis. In *2019 IEEE International Conference on Prognostics and Health Management*, 1–8. <https://doi.org/10.1109/ICPHM.2019.8819420>.
- [47] Patton, R. J., Uppal, F. J., Simani, S., & Polle, B. (2008). Reliable fault diagnosis scheme for a spacecraft attitude control system. *Journal of Risk and Reliability*, 22(2), 139–152. <https://doi.org/10.1243/1748006XJRR98>.
- [48] Patton, R. J., Uppal, F. J., Simani, S., & Polle, B. (2010). Robust FDI applied to thruster faults of a satellite system. *Control Engineering Practice*, 18(9), 1093–1109. <https://doi.org/10.1016/j.conengprac.2009.04.011>.

How to Cite: Simani, S., Ping Lam, Y., Farsoni, S., & Castaldi, P. (2024). Dynamic Neural Network Architecture Design for Predicting Remaining Useful Life of Dynamic Processes. *Journal of Data Science and Intelligent Systems*, 2(3), 141–152. <https://doi.org/10.47852/bonviewJDSIS3202967>