1 Nomenclature

$\boldsymbol{h}(k)$	k^{th} fused measurement, $k = 1, 2, \cdots, M$
$\boldsymbol{h}_j$	j^{th} fused variable, $j = 1, 2, \cdots, R$
$\boldsymbol{H}$	1-dimensional or multi-dimensional fusion result matrix consists of fused condition monitoring variables
M	number of measurements
R	number of fused variables
S	number of CM variables
$\boldsymbol{w}$	vector consists of linear fusion coefficients
$\boldsymbol{x}(k)$	k^{th} multi-dimensional measurement, $k = 1, 2, \cdots, M$
$\boldsymbol{x}_j$	j^{th} condition monitoring variable, $j = 1, 2, \cdots, S$
$\boldsymbol{X}$	data matrix consists of condition monitoring variables
AAKR	Auto-Associative Kernel Regression
AE	Auto-Encoder
BMU	Best Matching Unit
CM	Condition Monitoring
EoL	End of Life
FFT	Fast Fourier Transform
FPC	First Principal Component
FT	Failure Threshold
GPR	Gaussian Process Regression
GTM	Generative Topographic Mapping
HI	Health Indicator
IGBT	Insulated Gate Bipolar Transistor
KPCA	Kernel Principal Component Analysis
KT	Kernel Trick
LR	Logistic Regression
MAPE	Mean Absolute Percentage Error
PCA	Principal Component Analysis
PCR	Principal Component Regression
PEMFC	Proton Exchange Membrane Fuel Cell
PF	Particle Filter
PHI	Physical Health Indicator
PoF	Physics of Failure
RBF	Radial Basis Function
RMSE	Root Mean Squared Error

RUL	Remaining Useful Life
RVM	Relevance Vector Machine
SoH	State of Health
SOM	Self-Organizing Map
SNR	Signal-to-Noise Ratio
SVD	Singular Value Decomposition
SVM	Support Vector Machine
SVR	Support Vector Regression
TDNN	Time Delay Neural Network
VHI	Virtual Health Indicator

2 Introduction

3 Definition

Denote each CM variable as $\boldsymbol{x} \in \mathrm{R}^M$. Generally, it can be a time series, values of a probability density or a set of decisions. Let $\boldsymbol{X}$ be the matrix consists of all the CM variables, i.e., $\boldsymbol{X} = \left[\begin{array}{cccc} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \cdots & \boldsymbol{x}_S \end{array} \right] = \left[\begin{array}{cccc} \boldsymbol{x}(1); & \boldsymbol{x}(2); & \cdots; & \boldsymbol{x}(M) \end{array} \right] \in \mathrm{R}^{M\times S}$, where $\boldsymbol{x}_{1:S}$ are S CM variables. Let $\boldsymbol{H} = \left[\begin{array}{cccc} \boldsymbol{h}_1 & \boldsymbol{h}_2 & \cdots & \boldsymbol{h}_R \end{array} \right] = \left[\begin{array}{cccc} \boldsymbol{h}(1); & \boldsymbol{h}(2); & \cdots; & \boldsymbol{h}(M) \end{array} \right] \in \mathrm{R}^{M\times R}$ denote the fusion result in the same form with $\boldsymbol{X}$, and it can be a vector or a matrix.

In the most cases, $\boldsymbol{x}$ is a time series, i.e., each column $\boldsymbol{x}_j$, $j = 1, 2, \cdots, S$ of $\boldsymbol{X}$ denotes a time series characterizing the SoH, while each row $\boldsymbol{x}(k)$, $k = 1, 2, \cdots, M$ of it represents a multi-dimensional measurement collected at time t_k. The multi-dimensional time series here including both the directly collected CM signals from sensors, the time-domain features, the statistical features combined with a moving-window technique, and all other data in the form of multi-dimensional time series.

4 Architecture of Fusion Models

4.1 Linear Fusion

Linear fusion has been the most common model, it can be written as

$$\boldsymbol{H} = f(\boldsymbol{X}; \boldsymbol{w}) = \boldsymbol{X}\boldsymbol{w}, \tag{1}$$

where $\boldsymbol{w} = \left[\begin{array}{cccc} w_1 & w_2 & \cdots & w_S \end{array} \right] \in \mathrm{R}^S$ is a set of fusion coefficients.

4.2 Nonlinear Fusion

The linear fusion model is relatively simple in structure, and it has been widely used for its ability of generalization. On the contrary, the linear model is usually insufficient to accurately characterize the complicated relations between the multiple monitoring variables and the latent SoH, especially for complex and integrated systems [46]. As a result, complex nonlinear fusion models have attracted a lot of attention due to their powerful ability of interpreting the implicit structures.

4.2.1 Similarity-Based Fusion

General similarity metric includes distance, Pearson correlation coefficient, Kullback-Leibler (K-L) divergence, pointwise mutual information and kernel function etc.

Distance-Based Fusion The distance between two data points in the data space can characterize the similarity between them [5], and further distance means lower similarity, based on which idea it can effectively characterize the SoH and serve as an HI. The essential process of estimating the SoH is to comparing the deviation of the current operating state from the baseline operating state [8], i.e., the normal or healthy state. The greater the deviation is, i.e., in the data space, the further between the current collected data and that of the baseline, the worse the SoH is. As a result, the distance-based fusion methods have been widely applied on both data-level and feature-level, the essence of which is a nonlinear fusion model. Because the distance function is generally defined as a non-negative real number mapping, i.e., $dist : \mathcal{R}^S \to \mathcal{R}$, fusion function can be realized by calculate the distance between each row of $\boldsymbol{X}$ and the baseline, and the distance is adopted as an 1-dimensional fused HI, i.e., $\boldsymbol{H} = f(\boldsymbol{X}) = dist(\boldsymbol{x}(1:M), \boldsymbol{x}(*))$. $\boldsymbol{x}(*)$ denotes the measurement of the baseline state, which is specified as *priori* information, so it is regarded as a constant vector during the CM. Taking the 2-norm distance as an example, the fusion model is analyzed. The 2-norm distance (Euclidean distance) is defined as

$$\begin{aligned} h(k) &= Sim(\boldsymbol{x}(k), \boldsymbol{x}(*)) \\ &= Dist^E(\boldsymbol{x}(k), \boldsymbol{x}(*)) \\ &= \|\boldsymbol{x}(k) - \boldsymbol{x}(*)\|_2 \\ &= \sqrt{\sum_{j=1}^{S} [x_j(k) - x_j(*)]^2}. \end{aligned} \tag{2}$$

$Sim(\cdot,\cdot)$ is a row-wise operator measuring the similarity between rows of the input matrices. It can be seen from (2) that the current multi-dimensional measurement is fused by a nonlinear combination function into a non-negative real number.

Because of the measurement noise and other uncertainty, $\boldsymbol{x}(*)$ is commonly a random vector with the mean $\boldsymbol{\mu}(*)$ and the covariance matrix $\boldsymbol{\Sigma}(*)$, the Mahalanobis distance can be employed as the fusion function

$$\begin{aligned} h(k) &= dist^M(\boldsymbol{x}(k), \boldsymbol{x}(*)) \\ &= \sqrt{[\boldsymbol{x}(k) - \boldsymbol{\mu}(*)]\, \boldsymbol{\Sigma}^{-1}(*)\, [\boldsymbol{x}(k) - \boldsymbol{\mu}(*)]^{\mathrm{T}}}. \end{aligned} \tag{3}$$

There are various definitions of distance, including Manhattan distance, Minkowski distance, Hamming distance, Jaccard index, Levenshtein distance (edit distance), Jaro-Winkler distance, Lee distance, Hellinger distance, Canberra distance and Chebyshev distance etc.

Kernel Method The linear fusion model in (1) can be transformed into a nonlinear one by introducing a kernel function, which also measures the similarity between the inputs. Generally, for the kernel method, a mapping $\boldsymbol{\phi} : \mathcal{X} \to \mathcal{H}$ from the original data space $\mathcal{X} \subseteq \mathcal{R}^S$ to a reproducing Hilbert space (feature space) $\mathcal{H} \subseteq \mathcal{R}^{S^*}$ is constructed, where S^* is the number of dimensions of the mapped feature. The linkage between the vectors before and after mapping can be established by introducing a kernel function, i.e.,

$$\langle \boldsymbol{\phi}(\boldsymbol{x}_1), \boldsymbol{\phi}(\boldsymbol{x}_2) \rangle_{\mathcal{H}} = k(\boldsymbol{x}_1, \boldsymbol{x}_2), \tag{4}$$

where $\langle \cdot,\cdot \rangle_{\mathcal{H}} : \mathcal{H} \times \mathcal{H} \to \mathcal{R}$ is the standard inner production in $\mathcal{H}$, and $k(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \to \mathcal{R}$ is the kernel function, measuring the similarity between pair of inputs $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}$ [46, 25]. It can be seen from (1) that the mapping $\boldsymbol{\phi}(\cdot)$ is implicitly defined via the kernel function. According to Mercer's theorem, any symmetric positive semi-definite function can be defined as a kernel function, and an implicitly defined function $\boldsymbol{\phi}(\cdot)$ exists whenever the space $\mathcal{X}$ can be equipped with a suitable measure ensuring the function $k(\cdot,\cdot)$ satisfies Mercer's condition.

Then a linear combination can be constructed for the transformed vectors in the feature space for fusion

$$\boldsymbol{h}(k) = \boldsymbol{\phi}(\boldsymbol{x}(k))\, \boldsymbol{w} = f(\boldsymbol{x}(k); \boldsymbol{w}), \tag{5}$$

where $\boldsymbol{w} \in \mathcal{R}^{S^* \times 1}$. The fusion model $f(\cdot; \boldsymbol{w})$ is equivalent to the row-wise operator $\boldsymbol{\phi}(\cdot)\,\boldsymbol{w}$, and consequently, $f(\cdot; \boldsymbol{w})$ is a nonlinear fusion model due to the non-linearity of $\boldsymbol{\phi}(\cdot)$.

It is usually hard to compute $\boldsymbol{\phi}(\cdot)$ directly, so the Kernel Trick (KT) is introduced to indirectly build the fusion model. Set $\boldsymbol{\Phi} = \left[\ \boldsymbol{\phi}(\boldsymbol{t}(1));\ \ \boldsymbol{\phi}(\boldsymbol{t}(2));\ \ \cdots;\ \ \boldsymbol{\phi}(\boldsymbol{t}(N))\ \right] \in \mathcal{R}^{N \times S^*}$ and denote $\boldsymbol{w}$ as $\boldsymbol{w} = \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\beta}$, where $\boldsymbol{\beta} \in \mathcal{R}^{N \times 1}$, and $\boldsymbol{t}(n)$, $n = 1, 2, \cdots, N$ are all of the multi-dimensional CM measurements involved to training the fusion model (i.e., estimating $\boldsymbol{\beta}$). $\boldsymbol{t}(n)$ and $\boldsymbol{x}(k)$ have the same form and meaning. Then, the nonlinear fusion model (5) can be rewritten as

$$\begin{aligned}
\boldsymbol{h}(k) &= \boldsymbol{\phi}(\boldsymbol{x}(k))\,\boldsymbol{w} \\
&= \boldsymbol{\phi}(\boldsymbol{x}(k))\,\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\beta} \\
&= \begin{bmatrix} \boldsymbol{\phi}(\boldsymbol{x}(k))\,\boldsymbol{\phi}(\boldsymbol{t}(1))^{\mathrm{T}} & \boldsymbol{\phi}(\boldsymbol{x}(k))\,\boldsymbol{\phi}(\boldsymbol{t}(2))^{\mathrm{T}} & \cdots & \boldsymbol{\phi}(\boldsymbol{x}(k))\,\boldsymbol{\phi}(\boldsymbol{t}(N))^{\mathrm{T}} \end{bmatrix}\boldsymbol{\beta} \\
&= \begin{bmatrix} \langle\boldsymbol{\phi}(\boldsymbol{x}(k)),\boldsymbol{\phi}(\boldsymbol{t}(1))\rangle_{\mathcal{H}} & \langle\boldsymbol{\phi}(\boldsymbol{x}(k)),\boldsymbol{\phi}(\boldsymbol{t}(2))\rangle_{\mathcal{H}} & \cdots & \langle\boldsymbol{\phi}(\boldsymbol{x}(k)),\boldsymbol{\phi}(\boldsymbol{t}(M))\rangle_{\mathcal{H}} \end{bmatrix}\boldsymbol{\beta} \\
&= \begin{bmatrix} k(\boldsymbol{x}(k),\boldsymbol{t}(1)) & k(\boldsymbol{x}(k),\boldsymbol{t}(2)) & \cdots & k(\boldsymbol{x}(k),\boldsymbol{t}(N)) \end{bmatrix}\boldsymbol{\beta} \\
&= \boldsymbol{K}(\boldsymbol{x}(k),\boldsymbol{T})\,\boldsymbol{\beta}, \\
&= Sim(\boldsymbol{x}(k),\boldsymbol{T})\,\boldsymbol{\beta},
\end{aligned}$$

thus

$$\boldsymbol{h} = \boldsymbol{K}(\boldsymbol{X},\boldsymbol{T})\,\boldsymbol{\beta}, \tag{6}$$

where $\boldsymbol{K}(\cdot,\cdot)$ is a row-wise inner product operator for input matrices in $\mathcal{H}$, i.e.,

$$\boldsymbol{K}(\boldsymbol{X},\boldsymbol{T}) = \begin{bmatrix} k(\boldsymbol{x}(1),\boldsymbol{t}(1)) & k(\boldsymbol{x}(1),\boldsymbol{t}(2)) & \cdots & k(\boldsymbol{x}(1),\boldsymbol{t}(N)) \\ k(\boldsymbol{x}(2),\boldsymbol{t}(1)) & k(\boldsymbol{x}(2),\boldsymbol{t}(2)) & \cdots & k(\boldsymbol{x}(2),\boldsymbol{t}(N)) \\ \vdots & \vdots & \ddots & \vdots \\ k(\boldsymbol{x}(M),\boldsymbol{t}(1)) & k(\boldsymbol{x}(M),\boldsymbol{t}(2)) & \cdots & k(\boldsymbol{x}(M),\boldsymbol{t}(N)) \end{bmatrix}.$$

The fusion coefficients $\boldsymbol{\beta}$ in (6) can be solved via various methods, e.g., solving an optimization problem or analytically deriving based on other assumptions.

Compared (5) with (6), it can be seen that the parametric fusion model (5) is transformed into a non-parametric one, resulting in an increase of the amount of data to be stored. Besides, every time new information is fused using (6), the whole training set $\boldsymbol{T}$ will be involved, greatly increasing the calculation cost.

The selection of the kernel function will significantly impact the performance, and generally used kernel functions include linear kernel, polynomial kernel and Radial Basis Function (RBF) Gaussian kernel etc.

4.3 Neural Networks

4.3.1 Multi-Layer Perceptron

4.3.2 Recurrent neural network

4.3.3 Convolutional Neural Network

4.3.4 Self-Organizing Map

5 Solution of Fusion Models

5.1 Non-Optimization-Based Models

5.2 Optimization-Based Models

For the optimization-based fusion, the fusion model can be built by solving an optimization problem

$$f(\cdot) = \operatorname{argmin} m(f(\boldsymbol{X})), \tag{7}$$

where $m(\cdot)$ is the metric function measuring the performance of the fusion result in terms of the requirement of RUL prediction. For the sake of generality, the minimization is taken here as the example of the optimization problem.

5.2.1 Indirect-Optimization-Based Fusion

Fusion based on these metric functions aims to optimize the CM signal quality, the modeling accuracy and so on. This fusion method will not affect the prediction results directly, but will influence the prediction accuracy imperceptibly, which is indirect-optimization-based fusion. Lei et al. [21] reviewed several metrics, e.g., the monotonicity [31, 55, 16, 59, 4, 28, 3], robustness[59], trendability [16, 59, 20, 55, 15, 14, 3], identifiability [37, 55, 64], consistency [39, 37, 3], (modified) Mann–Kendal criterion [12], and some hybrid metrics [59, 37, 30, 32, 53, 11, 62].

Robustness and consistency are adopted to intuitively illustrate how the metrics above will have effect on RUL estimation. For an in-situ degraded device, under the condition of modeling the HI and the FT as a stochastic process and a stochastic variable respectively, the degradation model can be obtained by fitting the collected in-situ data, and the FT can be estimated as the mean of the CM measurements of the historical failed devices at their respective EoL, as implemented in [30, 32, 53]. Robustness is generally defined to characterize the random fluctuations in CM measurements, which is a result of the noisy environment, and it can be denoted as

$$Rob\left(\boldsymbol{x}\right) = Map\left(\left\|\boldsymbol{C}\left(\boldsymbol{x} - \hat{\boldsymbol{x}}\right)\right\|\right)$$

where $\boldsymbol{x}$ is a sample function of the stochastic degradation process, $\hat{\boldsymbol{x}}$ is the expectation, $\boldsymbol{C}$ is a diagonal matrix consists of a series of weighted coefficients tuning the importance of each measurement, $\|\cdot\|$ is the norm, and $Map\left(\cdot\right)$ is a mapping function. Under the same failure and operational mode, the degraded devices are supposed to exhibit some degree of similarity, e.g., the similarity of their FTs, which can be represented by the standard deviation of their FTs. As the FT is estimated by the measurement at EoL, the consistency can be denoted as

$$Con = Map\left(std\left(\boldsymbol{x}^i\left(EoL^i\right)\right)\right),\ i \in I,$$

where I is an index set consists of indices of all the degradation devices under the same failure and operational mode. In principle, a residual vector $(\boldsymbol{x} - \hat{\boldsymbol{x}})$ with a smaller norm represents a better robustness, while a smaller standard deviation of FTs under the same failure and operational mode represents a better consistency. Then the uncertainty of the RUL estimation can be significantly mitigated by optimizing the robustness of HI, as well as the consistency of FT, as shown in Fig. 1.

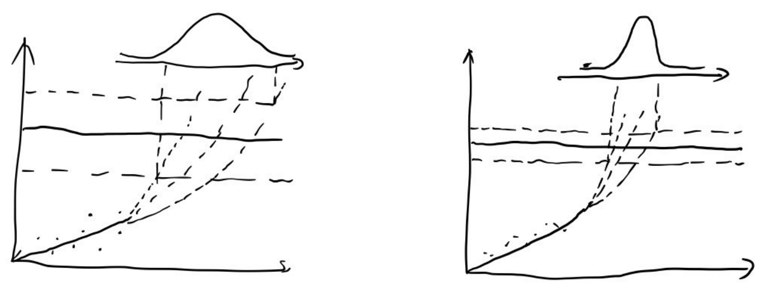

Figure 1: The effect of robustness and consistency on RUL estimation.

It can be concluded that the Indirect-Optimization-Based Fusion aims to optimize the fusion result under general metrics, which are potentially beneficial for RUL prediction. However, the performance of prognostics also depends on the degradation modeling, parameter estimation, and FT estimation, etc.

5.2.2 Direct-Optimization-Based Fusion

The prediction error is the most directly metric characterizing the performance of the used information. In this case, metric function $m\left(\cdot\right)$ can be taken as the error between the predicted RUL and the ground truth, i.e.,

$$f\left(\cdot\right) = \mathrm{argmin} m\left(\hat{RUL}\left(f\left(\boldsymbol{X}\right)\right); RUL\right).$$

where $\hat{RUL}$ is the estimator of RUL depending on the fusion result, and RUL is the corresponding actual one. Besides, Lei et al. [21] reviewed other metrics for RUL prediction, including, Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE) and $\alpha - \lambda$ accuracy, etc.

For this category of methods, the actual RUL is adopted as a response variable to guide the construction of fusion model, so as to guarantee the prognostic performance, and they are generally more robust when compared with Indirect-Optimization-Based Linear Fusion.

5.3 Optimization Methods

5.3.1 Bio-Inspired

Grey Wolf Optimizer Grey Wolf Optimizer is inspired by wolve's hunting behaviors including division and collaboration, which objective is to obtain an optimal set of hunting positions [62].

6 Literature Review

6.1 Non-Optimization-Based Linear Fusion

6.1.1 Non-Optimization-Based Adaptive Linear Fusion

Adaptive method can adapt to the complex and changeable environment, which can be denoted as

$$\boldsymbol{h}(k) = \sum_{j=1}^{S} w_j(\boldsymbol{x}(k))\, x_j(k), \tag{8}$$

where each fusion coefficient is a function of the current measurement.

Sun et al. [48] fused information from both capacitance, resistance and constant current charge time of lithium-ion batteries at data level. The 3 normalized CM signals were combined by a set of normalized weight coefficients. Besides the fusion coefficients can be updated at each monitoring time according to the last fusion result.

Based on a historical data base of a group of run-to-failure devices, Malinowski et al. [38] built a library of shapelets from all of the CM signals. For a monitored operating device, its collected CM signal were compared with the shapelets in the library, and some matching shapelets according to the Euclidean distance were selected. Such selection can be regarded as assigning to each shapelet in terms of the corresponding distance. If and only if the distance satisfies certain condition(s), the corresponding weight is not assigned as zero. This method is a special case of linear weighted combination with most of weight coefficients as zero, and it is sparse consequently.

In order to adaptively capture the local degradation changes conditioned to the feature population behavior, Atamuradov et al. [2] constructed an HI via an adaptive linear model, whose fusion coefficients were determined by the average distance between the corresponding CM feature and all the other features. Tse et al. [50] built a similar model, and the difference was that after the distance was calculated, it was mapped by an exponential function with a tuning parameter.

Cheng et al. [9] performed Singular Value Decomposition (SVD) to the CM data matrix $\boldsymbol{X}$, and then a linear fusion model was built for the multi-sensor signals, among which the weight coefficients were the corresponding normalized singular values.

At each CM time, Li et al. [23] calculated relative entropy between each pair of CM features, and the pair of features with the maximum relative entropy were combined in a linear way, until all of the features were fused into 1-dimension over time. The coefficients for each fusion step were determined by the probability distribution of the current value of the corresponding features.

Yan et al. [54] introduced the permutation entropy to determine the coefficients of a linear fusion model, which can measure the degree of the monotonicity, based on the assumption that the degradation components producing signals with larger trends contribute more to the deteriorated system.

6.2 Non-Optimization-Based Nonlinear Fusion

6.2.1 Distance-Based Methods

k-Nearest Neighbors Algorithm k-Nearest Neighbors Algorithm (k-NN) is a classical method which can deal with both classification and regression. When a new multi-dimensional measurement (in data space or feature space) is collected, it is compared with each member of a historical data base. The k nearest neighbor measurements in the historical data base will be selected according to the corresponding distance, and then the output will depend on that of the nearest neighbors.

Li et al. [27] used k-NN-based regression on the selected features to perform RUL prognostics for a railcar.

Other Distance-Based Fusion Models Li et al. [26] chose CM signals from four sensors as the Physical HIs (PHIs). Then the information were fused as the Euclidean distance between each multi-dimensional measurement and a predefined Failure Threshold (FT), i.e., the predicted last value from a regression model for each CM signal at the End of Life (EoL) of each degraded unit. Baraldi et. al [3] reconstructed a set of degradation features using the AAKR, and then a fused HI was constructed based on the Euclidean distance between the original features and the reconstructed ones. Yasmine et al. [43] used Mahalanobis distance to to fuse information on data-level as a data-driven precursor for a ship's deterioration. Zhao et al. [62] further fused features extracted by an enhanced auto-encoder. The last hidden layer of the enhanced auto-encoder was divided into several child modules, and then the output of each module was input to a corresponding SOM model. Distance between the input of each Self-Organizing Map (SOM) model and its Best Matching Unit (BMU) was adopted as a further fused HI.

Other definitions of distance can also be employed to fuse the information, e.g., 1-norm distance (Manhattan distance), infinity norm distance (Chebyshev distance) and the geodesic distance.

Chen et al. [8] fused the data of stack voltage, power and inner resistance based on the geodesic-distance-based fusion, which distance can effectively describe the spatial structure of the degradation information of Proton Exchange Membrane Fuel Cell (PEMFC).

Pan et al. [40] calculated Hotelling's T-squared statistic between the current and the healthy state, which is a kind of generalized squared interpoint distance, on the fused features to assess the degradation of wind turbine gearboxes.

6.2.2 Others

Zhou et al. [65] constructed an SoH classifier via Hidden Markov Model (HMM) on the selected 5 CM signals, so as to fuse them into a composite SoH, and then the long-term predicted SoH was used to estimate the RUL for aircraft engines.

6.3 Indirect-Optimization-Based Linear Fusion

For the linear case, the set of fusion coefficients $\boldsymbol{w}$ can be acquired as

$$\boldsymbol{w} = \text{argmax} m\left(\boldsymbol{X}\boldsymbol{w}\right). \tag{9}$$

6.3.1 Principal Component Analysis

Principal component analysis is one of the most widely used fusion methods whose procedure is to use an orthogonal transformation to fuse a set of possibly correlated CM variables into a set of linearly uncorrelated ones. Actually, it uses R sets of linear fusion models to fuse S CM variables into R variables ($R \leq S$). Let the R sets of linear fusion models be $f_i\left(\boldsymbol{X}\right) = f_i\left(\boldsymbol{x}_{1:S}\right) = \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \cdots & \boldsymbol{x}_S \end{bmatrix} \boldsymbol{w}_i = w_{i1}\boldsymbol{x}_1 + w_{i2}\boldsymbol{x}_2 + \cdots + w_{iS}\boldsymbol{x}_S$, $i = 1, 2, \cdots, R$, and they can be acquired by solving such a constraint optimization problem

$$f_i\left(\cdot\right) = \begin{cases} \text{argmax} Var\left(f_1\left(\boldsymbol{X}\right)\right), & i = 1, \\ \text{argmax} Var\left(f_i\left(\boldsymbol{X} - \sum_{k=1}^{i-1} f_k^{approx}\left(f_k\left(\boldsymbol{X}\right)\right)\right)\right), & i \geq 2, \end{cases} \tag{10}$$
$$\text{s.t.}\, Cov\left(f_i\left(\boldsymbol{X}\right), f_j\left(\boldsymbol{X}\right)\right) = 0,\ i \neq j,$$
$$\|\boldsymbol{w}_i\| = 1,\ i, j = 1, 2, \cdots, R.$$

where $f_k^{approx}\left(\cdot\right)$ is a reconstruction model corresponding to the k^{th} fusion model, i.e., $f_k^{approx}\left(\cdot\right) = \left[\cdot\right] \boldsymbol{w}_k^{\mathrm{T}}$. The fusion result $f_i\left(\cdot\right)$ is the so-called i^{th} Principal Components (PC). Generally, such data-level fusion methods with the form of dimensionality reduction lose information. PCA-based fusion tends to minimize the information loss under certain signal and noise models [29, 10], and it can be applied both on data level and feature level.

Wen et al. [52] selected 11 CM signals to monitor the State of Health (SoH) and predict the RUL of aircraft engines, and the selected signals were directly fused into an HI, i.e., the First Principal Component (FPC). Yu et al. [57] collected multiple CM signals for RUL prediction of batteries, including voltage, current and temperature, and then a set of statistical features for each CM signal were generated combined with a moving window, which consist of mean, variance, kurtosis, skewness, RMS and Shannon entropy. The total 18 features (3 CM signals $\times$ 6 statistical features) were then fused into 4 features using PCA. Liu et al. [36] extracted 6 features from the degradation data of cutting tools and then fused them into a 1-dimentional Health Indicator (HI), which can represent tool wear conditions. Zhang et al. [61] also employed the PCA-based fusion method to fuse 21 time- and frequency-domain features extracted from the noisy vibration signal into 3 composite features, so as to perform machine health Prognostic and Condition Monitoring (PCM).

6.3.2 General Indirect-Optimization-Based Linear Fusion Models

Under the assumption that 1) the degradation process is an irreversible process, and 2) the degradation processes of a set of degraded devices are supposed to exhibit a consistent pattern, Liu et al. [31] fused information on data-level by maximizing the monotonicity and minimizing the variance in the failure threshold of the fusion result jointly. They [32] also built an exponential model to describe the degradation process and they took the fitting accuracy of the model into consideration, constructing a data-level fusion model by optimizing the model fitting accuracy. Besides, they [30] further built a fusion model by optimizing a predefined a Signal-to-Noise Ratio (SNR) of the fusion result, simultaneously considering 1) the range of its variation, 2) the fitting accuracy of the exponential model, and 3) the variance in the failure threshold. Yan et al. [53] also built a data-level fusion model from the three perspectives above, and differently, they set the optimization objective as a linear weighted combination of these three metrics, to deal with the RUL prediction under multiple operational conditions. Chehade et al. [6] focused on better distinguishing different failure modes so as to acquire higher prognostic accuracy, so they fused data of aircraft engines by minimizing the hinge-loss. Inconsistency between the fused information from engines under different failure modes is consequently larger, better for RUL prediction under this condition. This inconsistency among devices under different failure modes or operating conditions is usually the opposite of the assumption of *consistency* (see Subsection 5.2.1 for details).

Run-to-failure data are commonly used for RUL prognostics, as a result, it is reasonable to assume that the degraded devices are completely healthy at the initial stage, while they are totally failed at their EoL [47]. Accordingly, Sun et al. [47] used a linear model to fuse the CM signals at data level into an normalized HI, which is with in $[0,1]$ so as to characterize the SoH, with optimizing the fused HI as 1 at the initial stage as well as 0 at the EoL. Jiang et al. [17] implemented linear fusion on data level with a similar criteria. They divided the degradation stage into a healthy stage and a faulty one, and then fused the selected CM signals into a normalized HI with optimizing it as 1 at the healthy stage, and 0 at the faulty stage.

Based on features fused via an SOM model, Zhao et al. [62] fused these features linearly and the fusion coefficients were obtained by optimizing a weighted sum of *monotonicity*, *trendability* and *prognosability* (see [3, 62] for detailed definitions).

6.4 Direct-Optimization-Based Linear Fusion

6.4.1 Principal Component Regression

Based on PCA, Principal Component Regression (PCR) is developed. The key point of this method is that the Principal Components (PCs) are adopted as independent variables for multiple linear regression.

6.4.2 General Direct-Optimization-Based Linear Fusion Models

Song et al. [45] built the linear fusion model at data level by minimizing the prediction error, they combined the fusion technique with a stochastic-process-based prediction method, and in this case, the fused HI has the analytical optimal property for RUL prediction. Kim et al. [19] improved the method of Song et al., i.e., they implemented the sensor selection by introducing a regularization term, with which the sparse fusion model merely involve informative sensor signals.

6.5 Indirect-Optimization-Based Nonlinear Fusion

6.5.1 Kernel-Optimization Combined methods

Under certain assumptions, the fusion coefficients of kernel-method-based fusion models, $\boldsymbol{w}$ in (5) or $\boldsymbol{\beta}$ in (6), can be derived by solving an optimization problem, e.g., Kernel Principal Component Analysis (KPCA), Support Vector Machine (SVM), Support Vector Regression (SVR), Gaussian Process Regression (GPR) and Relevance Vector Machine (RVM) etc.

Kernel Principal Component Analysis Under certain signal and noise models, information is fused via PCA with the minimum information loss, and it has been one of the most widely-used fusion methods. However, most of the signals acquired from industrial systems are non-linear and non-stationary [44]. Restricted by the linear structure of the fusion model, PCA may not characterize some latent relations within complicated devices, so it is improved by being combined with kernel method, forming Kernel Principal Component Analysis (KPCA).

In principle, the main differences between PCA and KPCA are: 1) the original data matrix $\boldsymbol{X}$ in (10) is substituted by the mapped feature matrix $\boldsymbol{\Phi}$ (see (4.2.1) for the definition); 2) the kernel trick is introduced to perform standardization and eigendecomposition. Because of the introduction of the nonlinear implicit mapping $\boldsymbol{\phi}(\cdot)$ (see (4.2.1) for the definition), the characterization ability of the fusion model has been significantly improved, and it is expected to perform better.

Duan et al. [11] extracted and selected 8 cumulative features from the vibration signal of bearings, and then these features were fused via KPCA for better describing the signal characteristics from different aspects.

Chen et al. [7] performed a data-level fusion based on KPCA which was used to fuse the original CM signals directly, and then the multi-dimensional fused features were output to be be further processed.

Pan et al. [40] used KPCA on the reconstructed signals.

Gaussian Process Regression For GPR, the fusion result is *a priori* assumed as a Gaussian process, and then the fused HI is an element within the family of finite-dimensional distributions of the Gaussian process, specifically, a multivariate normal distribution with the mean $\boldsymbol{h}$, and the covariance matrix $\boldsymbol{K}$, which is the matrix of kernel functions. GPR is essentially equivalent to a RBF network [65]. The mean function and the variance of the fusion result can be respectively derived according to the fusion model and Bayes' theorem. The fusion coefficients $\boldsymbol{\beta}$ can be derived based on the idea of ridge regression, by minimizing the residual as well as the L^2 regularization term.

Li et al. [25] extracted 11 CM features for batteries, and then they implemented a feature-level fusion on the selected 6 features via GPR, and a squared exponential covariance kernel function was selected. The selected features were fused to fit the measurements of a PHI that can well characteristic the actual SoH of the batteries.

Support Vector Machine and Support Vector Regression Some researches using Support Vector Machine (SVM) to distinguish SoH as a series of discrete states, and the probability of being in each SoH is adopted as the fused HI, in terms of which the long-term prognostics can be achieved. For SVM, the kernel method is combined with minimizing the hinge-loss of the fusion result, so it is widely used as a classifier.

Kim et al. [18] extracted 3 statistical features from the vibration signal of bearings of High Pressure-Liquefied Natural Gas (HP-LNG) pumps and they set 5 SoH characterizing the degradation level. Then SVM was adopted to fuse those features as probabilities, respectively with which, the degraded device is in the corresponding SoH.

SVR is similar to SVM. For regression, the commonly used residual sum of squares is substituted by the hinge-loss for being minimized.

Wang et al. [51] adopted SVR to capture the implicit relation between voltage, current, operated cycles and capacity, which is a commonly used PHI for batteries.

Kernel Adaptive Filter GPR, SVR, SVM and RVM are offline trained and derived in batch setting. However, for RUL prediction, a typical case is that multiple measurements arrive sequentially. In these batch methods, the built model have to be retrained for updating. To this end, Kernel Adaptive Filter (KAF) is proposed to incrementally update the constructed regression model. For example, for Kernel Recursive Least Squares method, the kernel least square is trained on collected data, forming an initial filter model, and a iterative format is derived. As a new measurement sample arrives, the regularized predictive residual error sum of squares is minimized in an iterative way, which will result in a growing size of the model. In order to avoid the continuous increase in memory and time cost as CM measurements are input, criteria [41, 42, 13, 34, 63] for online sparsification are introduced. According to these techniques, only new measurements that satisfy a specific criterion are taken into consideration to build the consequently sparse model, and the others will be discarded. Zhou et al. [65] further improved this technique by carrying out the structure update and coefficients adjustment independently, so as to fully use information in redundant samples.

6.5.2 Auto-Encoder and Its Variants

Zhao et al. [62] developed an enhanced auto-encoder by adding to automatically fuse highly abstracted features extracted via Fast Fourier Transform (FFT). For enhanced auto-encoder, an additional regularization term was added to the optimization objective which served the dual purposes of signal reconstruction and enhanced predictability.

6.5.3 AI approaches

In order to avoid the determination of precise segments or feature boundaries, and adapt to the dynamic time-domain changes in the signal, Alghassi [1] et al. fused a physical precursor (collector emitter voltage) and its increment for Insulated Gate Bipolar Transistors (IGBTs) via a Time Delay Neural Network (TDNN).

6.5.4 Other Indirect-Optimization-Based Nonlinear Fusion Models

Song et al. [46] combined the kernel method and the SNR proposed by Liu et al. [30]. They set the SNR as the the objective of optimization, as well as a second-order polynomial kernel as the kernel function, to build a kernel-method-based nonlinear fusion model. Under the same objective and with the same RUL prediction model, the nonlinear fusion model shows better performance than the linear model on characterizing the latent relation between information sources for complicated systems, with which the RUL is predicted more accurately.

If the failed and healthy states of run-to-failure devices in a historical data base are distinguished, a nonlinear fusion model can be developed via Logistic Regression (LR). In this case, the fused HI is the probability of the event *failure*. The sigmoid function is introduce to map the linear combination of multi-dimensional variables into $[0, 1]$, as well as maximizing the log-likelihood probability. Yu et al. [58, 56] developed a nonlinear fusion model based on LR, fusing original data into a statistical HI. Then the variation pattern of HI versus time was also determined via LR for long-term prognostics.

To fully use information from both two Structural Health Monitoring (SHM) techniques, i.e., acoustic emission and digital image correlation, Eleftheroglou et al. [12] built a nonlinear fusion model by summing the weighted product of pairs of features from those two sources. The weighted coefficients were solved by maximizing the modified Mann–Kendal criterion of the fusion result, which can also reflect its monotonicity.

6.6 Direct-Optimization-Based Nonlinear Fusion

6.6.1 Kernel-Optimization Combined methods

Relevance Vector Machine For RVM, the fusion coefficients $\boldsymbol{\beta}$ are further assumed as a mutually independent multivariate normal distribution with the mean vector $\mathbf{0}$. The covariance of $\boldsymbol{\beta}$ can be determined by literately maximizing the marginal likelihood. During the iterations, most components of $\boldsymbol{\beta}$ tend to 0, resulting in a sparse fusion model. Finally, the mean and the variance of the fusion result can also be derived respectively according to the fusion model and Bayes' theorem.

Support Vector Regression Li et al. [27] extracted 248 features from wheel impact load detector, machine vision systems, and optical geometry detectors of a railcar, to predict the RUL of its wheels and trucks (bogies). All the selected features were fused via SVR and the actual RUL values were set as the response variable.

Gaussian Process Regression Liu et al. [33] performed GPR for RUL prediction both on a simulation data set of aircraft engines and a practical data set of a flyable electro-mechanical actuator.

6.6.2 AI approaches

Chen et al. [7] performed a feature-level fusion based on Gated Recurrent Unit (GRU) based Recurrent Neural Network (RNN) for the fused features, which were extracted via KPCA.

6.6.3 Others

Li et al. [27] used Decision Trees (DT) to perform a feature-level fusion for RUL prediction of components of a railcar. The feature space was partitioned as multiple mutually exclusive cells, and the predicted value would depend on the cell that the corresponding measurement is in. RUL values were set as the response variable of the DT-based regression method.

7 Data Imperfection

LOF [9]

Sensor anomaly is one of the main reasons for data imperfection. Liu et al. [33] took this case into consideration, and they detected the anomalous sensors as well as recovering them via Least Square Support Vector Machine (LS-SVM). From two case studies, it can be observed that the recovered data from the anomalous sensors could also have positive impact on prediction.

8 Model Fusion

9 Model-Level and Decision-Level Fusion

9.1 Serial Structure Fusion

Zhou et al. [48] analyzed the degradation process of lithium-ion batteries and constructed a PHI, i.e., mean voltage falloff. The PHI was then output to RVM for regression so as to characterize the latent information in the structure of the PHI, in which process, information from physical property and the internal structure of data is fused.

Liu et al. [35] extracted and selected several features via Hilbert-Huang Transform (HHT) from vibration signals of bearings. The selected features were combined with a modified physical degradation model (Paris' Law), so as to introduce information from such physical process into PF as the state transition function to perform long-term RUL prediction.

Zhou et al. [65] selected 5 sensors to monitor aero-engines, and for each sensor, they proposed a reduced kernel recursive least squares (RKRLS) algorithm to predict its long-time future values in an iterative way. The value of one-step prediction was realized by combining several previous measurements, to acquire temporal information. Then the predicted time series were input to HMM for RUL estimation, in which the mutual information between different sensors was captured as well.

9.2 Parallel Structure Fusion

Yasmine et al. [43] employed the Bayesian Network to perform a decision-level fusion, fusing the information from a data-driven model and a Physics-of-Failure (PoF) model for the SoH monitoring of a ship.

Malinowski et al. [38] built a library of shapelets on a run-to-failure training set, and each shapelets convey information about RUL. Then the RUL of a degrading device was determined by the mean of information from shapelets that were matched with it.

Li et al. [25] set the SoH represented by a PHI as the response variable, and the 6 selected features as independent variables, to realize regression via GPR.

Li et al. [27] performed a parallel decision-level fusion by averaging all of the predicted RUL values from multiple CART trees, which were individual predictors.

Li et al. [22] proposed a linear fusion model, to fuse two transformed lifetime distribution respectively estimated using data from both simulation and field lifetime collection.

Yu et al. [58] used LR to express the relation of a fused HI versus operated time, which is equivalent to describing the law of its evolution by an empirical model, i.e., sigmoid function, and then the fitted empirical model was input to PF as the state transition function.

Liu et al. [33] used a nonlinear degradation auto-regressive model to acquire information of degradation trend contained in data, and then it was input to a Regularized Particle Filter (RPF). At the same time, an empirical degradation model was adopted as the state transition function, so as to introduce information from experts' experience. These two categories of information were fused via RPF by taking the output of the auto-regressive model as future measurements.

Alghassi et al. [1] introduced information from experts' knowledge to determine the failure threshold of IGBTs.

9.3 Tree Structure Fusion

Zhang et al. [60] built a regression model based on Relevance Vector Machine (RVM), which uses Bayesian inference and kernel method. The solution of RVM is sparse, meaning the regression model is built on a subset of the training set, and the selected data are called relevance vectors [49]. In [60], the capacity of lithium-ion batteries is adopted as the CM signal, based on which the RVM-based regression is implemented and the relevance vectors are output. The relevance vectors (i.e., the selected CM measurements) are used to fitted a proposed empirical degradation model, which is then output as the initial model to a Particle Filter (PF) predictor for RUL prognostics. Because such empirical models commonly come from experts' knowledge, the information from experts and data are consequently fused.

10 Hybrid Fusion

10.1 Ensemble Learning

Ensemble learning is a typical structure of hybrid fusion, which commonly consists of individual learners and a global decision module. Information at data-level and feature-level is fused by the individual learners, and then a global decision is made by the global decision module, which is parallel-decision-level fusion. It should be mentioned that if all of the individual learners only process information from identical source, then the structure will degenerate to the parallel-decision-level fusion.

Li et al. [27] performed feature-level and decision level hybrid fusion via the ensemble learning algorithms, i.e., Random Forests (RF) regression and Quantile Regression Forests. Each individual learner in RF was a CART tree, which received information from multi-dimensional feature space and output a predicted value. Then the outputs from all of the CART trees are averaged by the decision module as the global prediction.

10.2 Others

Yu et al. [57] combined the Generative Topographic Mapping (GTM) and Bayesian-Inference Probability (BIP) to further fuse 4 fused features obtained by PCA, into a 1-dimentional HI.

Yasmine et al. [43] performed a hybrid fusion based on data-level fusion and decision-level fusion to monitor the SoH for a ship. They employed the Mahalanobis distance to fuse information on data level so as to build a data-driven HI, and they also predict the RUL for the ship using a Physics-of-Failure (PoF) model based on its iron structures deterioration. Then they used a Bayesian network to implement a decision-level fusion combining the information from those two sources.

Chen et al. [7] performed a data-feature-level hybrid fusion based on KPCA and Gated Recurrent Unit (GRU) based Recurrent Neural Network (RNN). First, KPCA was used to fuse the information of original CM signals directly, and then the multi-dimensional fused features were input to GRU to be further fused at feature level.

Li et al. [24] also combined KPCA and GRU, and they only adopted the improved FPC by exponentially weighted moving average as the HI. In this way, the random fluctuations in the fusion result, which may still interfere with RUL estimation.

Malinowski et al. [38] performed a data-level and a decision-level fusion via shapelets matching and average method respectively.

Yu et al. [58] performed a data-level fusion via LR, and a state transition function was fitted based on the fused HI. Then the fused HI and the state transition function were input to PF for long-term prediction so as to estimate the RUL.

Li et al. [22] fused information from multiple categories of sources including experts' knowledge, physical process and data. Besides, for information from data, there are two categories of data sources, that from finite element simulation and lifetime transformed stress. Firstly, the lifetime was transformed to the riskiest point's stress, which was regarded as a PHI, and then distribution of the riskiest point's stress was estimated. This estimation was performed on data from both simulation and collection, during this process, the experts' experience was introduced to quantify the credibility of the two respectively fitted distribution. Finally, the two fitted distributions were fused based on the quantified credibility using a proposed linear model.

Alghassi et al. [1] performed a data-level fusion via TDNN to fuse collector emitter voltage and its increment of IGBTs as a degradation precursor, and based on which as well as a failure threshold defined according to experts' knowledge, RUL prediction for IGBTs was achieved.

References

[1] Alireza Alghassi, Mohammad Samie, and Suresh Perinpanayagam. Stochastic rul calculation enhanced with tdnn-based igbt failure modeling. *IEEE Transactions on Reliability*, 65(2):558–573, 2016.

[2] V. Atamuradov, K. Medjaher, F. Camci, P. Dersin, and N. Zerhouni. Railway point machine prognostics based on feature fusion and health state assessment. *Ieee Transactions on Instrumentation and Measurement*, 68(8):2691–2704, 2019.

[3] P Baraldi, G Bonfanti, and E Zio. Differential evolution-based multi-objective optimization for the definition of a health indicator for fault diagnostics and prognostics. *Mechanical Systems and Signal Processing*, 102:382–400, 2018.

[4] F. Camci, K. Medjaher, N. Zerhouni, and P. Nectoux. Feature evaluation for effective bearing prognostics. *Quality and Reliability Engineering International*, 29(4):477–486, 2013.

[5] Abdallah Chehade, Scott Bonk, and Kaibo Liu. Sensory-based failure threshold estimation for remaining useful life prediction. *IEEE Transactions on Reliability*, 66(3):939–949, 2017.

[6] Abdallah Chehade, Changyue Song, Kaibo Liu, Abhinav Saxena, and Xi Zhang. A data-level fusion approach for degradation modeling and prognostic analysis under multiple failure modes. *Journal of Quality Technology*, 50(2):150–165, 2018.

[7] J. L. Chen, H. J. Jing, Y. H. Chang, and Q. Liu. Gated recurrent unit based recurrent neural network for remaining useful life prediction of nonlinear deterioration process. *Reliability Engineering & System Safety*, 185:372–382, 2019.

[8] Jiayu Chen, Dong Zhou, Chuan Lyu, and Chen Lu. A novel health indicator for pemfc state of health estimation and remaining useful life prediction. *International Journal of Hydrogen Energy*, 42(31):20230–20238, 2017.

[9] C. Cheng, J. H. Wang, W. X. Teng, M. L. Gao, B. C. Zhang, X. J. Yin, and H. Luo. Health status prediction based on belief rule base for high-speed train running gear system. *Ieee Access*, 7:4145–4159, 2019.

[10] Gustavo Deco and Dragan Obradovic. *An information-theoretic approach to neural computing*. Springer, 1996.

[11] Lixiang Duan, Fei Zhao, Jinjiang Wang, Ning Wang, and Jiwang Zhang. An integrated cumulative transformation and feature fusion approach for bearing degradation prognostics. *Shock and Vibration*, 2018.

[12] N. Eleftheroglou, D. Zarouchas, T. Loutas, R. Alderliesten, and R. Benedictus. Structural health monitoring data fusion for in-situ life prognosis of composite structures. *Reliability Engineering & System Safety*, 178:40–54, 2018.

[13] Y. Engel, S. Mannor, and R. Meir. The kernel recursive least-squares algorithm. *Ieee Transactions on Signal Processing*, 52(8):2275–2285, 2004.

[14] Qinghua Hu, Xunjian Che, Lei Zhang, David Zhang, Maozu Guo, and Daren Yu. Rank entropy-based decision trees for monotonic classification. *Ieee Transactions on Knowledge and Data Engineering*, 24(11):2052–2064, 2012.

[15] Qinghua Hu, Weiwei Pan, Lei Zhang, David Zhang, Yanping Song, Maozu Guo, and Daren Yu. Feature selection for monotonic classification. *Ieee Transactions on Fuzzy Systems*, 20(1):69–81, 2012.

[16] Kamran Javed, Rafael Gouriveau, Noureddine Zerhouni, and Patrick Nectoux. Enabling health monitoring approach based on vibration data for accurate prognostics. *Ieee Transactions on Industrial Electronics*, 62(1):647–656, 2015.

[17] Wei Jiang, Jianzhong Zhou, Yang Zheng, and Han Liu. A hybrid degradation tendency measurement method for mechanical equipment based on moving window and grey-markov model. *Measurement Science and Technology*, 28(11), 2017.

[18] Hack-Eun Kim, Andy C. C. Tan, Joseph Mathew, and Byeong-Keun Choi. Bearing fault prognosis based on health state probability estimation. *Expert Systems with Applications*, 39(5):5200–5213, 2012.

[19] Minhee Kim, Changyue Song, and Kaibo Liu. A generic health index approach for multisensor degradation modeling and sensor selection. *IEEE Transactions on Automation Science and Engineering*, PP(99):1–12, 2019.

[20] Yaguo Lei, Naipeng Li, Szymon Gontarz, Jing Lin, Stanislaw Radkowski, and Jacek Dybala. A model-based method for remaining useful life prediction of machinery. *IEEE Transactions on Reliability*, 65(3):1314–1326, 2016.

[21] Yaguo Lei, Naipeng Li, Liang Guo, Ningbo Li, Tao Yan, and Jing Lin. Machinery health prognostics: A systematic review from data acquisition to rul prediction. *Mechanical Systems and Signal Processing*, 104:799–834, 2018.

[22] He Li, Hong-Zhong Huang, Yan-Feng Li, Jie Zhou, and Jinhua Mi. Physics of failure-based reliability prediction of turbine blades using multi-source information fusion. *Applied Soft Computing*, 72:624–635, 2018.

[23] Hongru Li, Jian Sun, Hui Ma, Zaike Tian, and Yifan Li. A novel method based upon modified composite spectrum and relative entropy for degradation feature extraction of hydraulic pump. *Mechanical Systems and Signal Processing*, 114:399–412, 2019.

[24] X. Q. Li, H. K. Jiang, X. Xiong, and H. D. Shao. Rolling bearing health prognosis using a modified health index based hierarchical gated recurrent unit network. *Mechanism and Machine Theory*, 133:229–249, 2019.

[25] Xiaoyu Li, Zhenpo Wang, and Jinying Yan. Prognostic health condition for lithium battery using the partial incremental capacity and gaussian process regression. *Journal of Power Sources*, 421:56–67, 2019.

[26] Yongxiang Li, Jianming Shi, Gong Wang, and Mengying Zhang. An ensemble model for engineered systems prognostics combining health index synthesis approach and particle filtering. *Quality and Reliability Engineering International*, 33(8):2711–2725, 2017.

[27] Zhiguo Li and Qing He. Prediction of railcar remaining useful life by multiple data source fusion. *Ieee Transactions on Intelligent Transportation Systems*, 16(4):2226–2235, 2015.

[28] Linxia Liao. Discovering prognostic features using genetic programming in remaining useful life prediction. *Ieee Transactions on Industrial Electronics*, 61(5):2464–2472, 2014.

[29] R. Linsker. Self-organization in a perceptual network. *Computer*, 21(3):105–117, 1988.

[30] Kaibo Liu, Abdallah Chehade, and Changyue Song. Optimize the signal quality of the composite health index via data fusion for degradation modeling and prognostic analysis. *IEEE Transactions on Automation Science and Engineering*, 14(3):1504–1514, 2017.

[31] Kaibo Liu, Nagi Z Gebraeel, and Jianjun Shi. A data-level fusion model for developing composite health indices for degradation modeling and prognostic analysis. *IEEE Transactions on Automation Science and Engineering*, 10(3):652–664, 2013.

[32] Kaibo Liu and Shuai Huang. Integration of data fusion methodology and degradation modeling process to improve prognostics. *IEEE Transactions on Automation Science and Engineering*, 13(1):344–354, 2016.

[33] Liansheng Liu, Qing Guo, Datong Liu, and Yu Peng. Data-driven remaining useful life prediction considering sensor anomaly detection and data recovery. *Ieee Access*, 7:58336–58345, 2019.

[34] Weifeng Liu, Il Park, and Jose C. Principe. An information theoretic approach of designing sparse kernel adaptive filters. *Ieee Transactions on Neural Networks*, 20(12):1950–1961, 2009.

[35] X. J. Liu, P. Song, C. Yang, C. B. Hao, and W. J. Peng. Prognostics and health management of bearings based on logarithmic linear recursive least-squares and recursive maximum likelihood estimation. *Ieee Transactions on Industrial Electronics*, 65(2):1549–1558, 2018.

[36] Yingchao Liu, Xiaofeng Hu, and Wenjuan Zhang. Remaining useful life prediction based on health index similarity. *Reliability Engineering & System Safety*, 185:502–510, 2019.

[37] Zhiliang Liu, Ming J. Zuo, and Yong Qin. Remaining useful life prediction of rolling element bearings based on health state assessment. *Proceedings of the Institution of Mechanical Engineers Part C-Journal of Mechanical Engineering Science*, 230(2):314–330, 2016.

[38] Simon Malinowski, Brigitte Chebel-Morello, and Noureddine Zerhouni. Remaining useful life estimation based on discriminating shapelet extraction. *Reliability Engineering & System Safety*, 142:279–288, 2015.

[39] A. Mosallam, K. Medjaher, and N. Zerhouni. Data-driven prognostic method based on bayesian approaches for direct remaining useful life prediction. *Journal of Intelligent Manufacturing*, 27(5):1037–1048, 2016.

[40] Yubin Pan, Rongjing Hong, Jie Chen, Jaskaran Singh, and Xiaodong Ji. Performance degradation assessment of a wind turbine gearbox based on multi-sensor data fusion. *Mechanism and Machine Theory*, 137:509–526, 2019.

[41] John Platt. A resource-allocating network for function interpolation. *Neural Computation*, 3(2):213–225, 1991.

[42] Cedric Richard, Jose Carlos M. Bermudez, and Paul Honeine. Online prediction of time series data with kernels. *Ieee Transactions on Signal Processing*, 57(3):1058–1067, 2009.

[43] Yasmine Z. Rosunally, Stoyan Stoyanov, Chris Bailey, Peter Mason, Sheelagh Campbell, George Monger, and Ian Bell. Fusion approach for prognostics framework of heritage structure. *Ieee Transactions on Reliability*, 60(1):3–13, 2011.

[44] B. Scholkopf, A. Smola, and K. R. Muller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

[45] Changyue Song and Kaibo Liu. Statistical degradation modeling and prognostics of multiple sensor signals via data fusion: A composite health index approach. *IISE Transactions*, pages 1–15, 2018.

[46] Changyue Song, Kaibo Liu, and Xi Zhang. Integration of data-level fusion model and kernel methods for degradation modeling and prognostic analysis. *IEEE Transactions on Reliability*, 67(2):640–650, 2018.

[47] Jianzhong Sun, Hongfu Zuo, Wenbin Wang, and Michael G. Pecht. Application of a state space modeling technique to system prognostics based on a health index for condition-based maintenance. *Mechanical Systems and Signal Processing*, 28:585–596, 2012.

[48] Yongquan Sun, Xueling Hao, Michael Pecht, and Yapeng Zhou. Remaining useful life prediction for lithium-ion batteries based on an integrated health indicator. *Microelectronics Reliability*, 88-90:1189–1194, 2018.

[49] Michael E Tipping. Sparse bayesian learning and relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[50] Y. L. Tse, M. E. Cholette, and P. W. Tse. A multi-sensor approach to remaining useful life estimation for a slurry pump. *Measurement*, 139:140–151, 2019.

[51] Fu-Kwun Wang and Tadele Mamo. A hybrid model based on support vector regression and differential evolution for remaining useful lifetime prediction of lithium-ion batteries. *Journal of Power Sources*, 401:49–54, 2018.

[52] Pengfei Wen, Shaowei Chen, Shuai Zhao, Yong Li, Yan Wang, and Zhi Dou. A novel bayesian update method for parameter reconstruction of remaining useful life prognostics. In *2019 IEEE International Conference on Prognostics and Health Management (ICPHM)*. IEEE.

[53] Hao Yan, Kaibo Liu, Xi Zhang, and Jianjun Shi. Multiple sensor data fusion for degradation modeling and prognostics under multiple operational conditions. *Ieee Transactions on Reliability*, 65(3):1416–1426, 2016.

[54] Shufa Yan, Biao Ma, Changsong Zheng, and Jianhua Chen. An optimal lubrication oil replacement method based on selected oil field data. *Ieee Access*, 7:92110–92118, 2019.

[55] Feng Yang, Mohamed Salahuddin Habibullah, Tianyou Zhang, Zhao Xu, Pin Lim, and Sivakumar Nadarajan. Health index-based prognostics for remaining useful life predictions in electrical machines. *Ieee Transactions on Industrial Electronics*, 63(4):2633–2644, 2016.

[56] J. B. Yu. Tool condition prognostics using logistic regression with penalization and manifold regularization. *Applied Soft Computing*, 64:454–467, 2018.

[57] Jianbo Yu. State-of-health monitoring and prediction of lithium-ion battery using probabilistic indication and state-space model. *Ieee Transactions on Instrumentation and Measurement*, 64(11):2937–2949, 2015.

[58] Jianbo Yu. Aircraft engine health prognostics based on logistic regression with penalization regularization and state-space-based degradation framework. *Aerospace Science and Technology*, 68:345–361, 2017.

[59] Bin Zhang, Lijun Zhang, and Jinwu Xu. Degradation feature selection for remaining useful life prediction of rolling element bearings. *Quality and Reliability Engineering International*, 32(2):547–554, 2016.

[60] Y. Z. Zhang, R. Xiong, H. W. He, and M. Pecht. Validation and verification of a hybrid method for remaining useful life prediction of lithium-ion batteries. *Journal of Cleaner Production*, 212:240–249, 2019.

[61] Yang Zhang, Paul Hutchinson, Nicholas A. J. Lieven, and Jose Nunez-Yanez. Adaptive event-triggered anomaly detection in compressed vibration data. *Mechanical Systems and Signal Processing*, 122:480–501, 2019.

[62] Lin Zhao and Xue Wang. A deep feature optimization fusion method for extracting bearing degradation features. *Ieee Access*, 6:19640–19653, 2018.

[63] Songlin Zhao, Badong Chen, Zheng Cao, Pingping Zhu, and Jose C. Principe. Self-organizing kernel adaptive filtering. *Eurasip Journal on Advances in Signal Processing*, 2016.

[64] Xiaomin Zhao, Ming J. Zuo, Zhiliang Liu, and Mohammad R. Hoseini. Diagnosis of artificially created surface damage levels of planet gear teeth using ordinal ranking. *Measurement*, 46(1):132–144, 2013.

[65] Haowen Zhou, Jinquan Huang, and Feng Lu. Reduced kernel recursive least squares algorithm for aero-engine degradation prediction. *Mechanical Systems and Signal Processing*, 95:446–467, 2017.

Table 1: Literature review of data-level and feature-level fusion

Solution \ Structure		Linear fusion		Nonlinear fusion		
Non-optimization-based	Stationary		Malinowski et al. [38]$^{-m}$ Li et al. [27]$^{-m}$ Li et al. [22]$^{-m}$	Distance-based		Li et al. [26]^{d} Baraldi et. al [3]^{f} Yasmine et al. [43]^{d} Chen et al. [8]^{d} Li et al. [27]$^{f-}$ Zhao et al. [62]^{f} Pan et al. [40]$^{-f}$
	Adaptive		Sun et al. [48]^{d} Malinowski et al. [38]$^{d-}$ Atamuradov et al. [2]^{f} Cheng et al. [9]^{d} Li et al. [23]^{d} Tse et al. [50]^{f} Yan et al. [54]^{d}	Others		Yu et al. [57]$^{-f}$ Zhou et al. [65]^{d}
Optimization-based	Indirect	PCA	Yu et al. [57]$^{d-}$ Wen et al. [52]^{d} Liu et al. [36]^{f} Zhang et al. [61]^{f}	Kernel method	KPCA	Duan et al. [11]^{f} Chen et al. [7]$^{d-}$ Li et al. [24]^{f} Pan et al. [40]$^{d-}$
					GPR	Li et al. [25]$^{f-}$
		General model	Liu et al. [31, 32, 30]^{d} Yan et al. [53]^{d} Sun et al. [47]^{d} Jiang et al. [17]^{d} Chehade et al. [6]^{d} Zhao et al. [62]^{f}		SVM	Kim et al. [18]^{f}
					SVR	Wang et al. [51]^{d}
					General model	Song et al. [46]^{d}
				AI approaches	AE	Zhao et al. [62]^{f}
					TDNN	Alghassi et al. [1]$^{d-}$
				Others		Yu et al. [58]$^{d-}$ Yu et al. [56]^{d} Eleftheroglou et al. [12]^{f}
	Direct	PCR	Li et al. [27]$^{f-}$	SVR		Li et al. [27]$^{f-}$
				GPR		Liu et al. [33]^{d}
		General model	Song et al. [45]^{d} Kim et al. [19]^{d}	AI approaches		Chen et al. [7]$^{-f}$
				Others		Li et al. [27]$^{f-}$

d: data-level fusion
f: feature-level fusion
m: model-level or decision-level fusion
$*-*$: hybrid fusion