Wavelet Scattering Network Based Bearing Fault Detection

Taoufik Bourgana[1], Robert Brijder[1], Ted Ooijevaar[1], and Agusmian Partogi Ompusunggu[1]

[1] *Flanders Make vzw, CoreLab DecisionS, Leuven, 3001, Belgium*
taoufik.bourgana@flandersmake.be
robert.brijder@flandersmake.be
ted.ooijevaar@flandersmake.be
agusmian.ompusunggu@flandersmake.be

ABSTRACT

This paper describes an algorithm for bearing fault detection using wavelet scattering networks (WSNs) as a preprocessing step for feature space generation. WSNs are based on the wavelet transform, where translation-invariant features are computed that are locally stable to deformation, making it particularly suitable for classification and clustering purposes. The method is experimentally validated with data acquired during 70 accelerated lifetime runs of bearings on 7 identical setups and compared to: (1) various statistical features such as root mean square, kurtosis, crest factor, and peak value, (2) a squared envelope spectrum method, (3) a supervised learning method based on convolutional neural networks. This comparison uses industrially relevant performance metrics, taking into account accuracy (including the classic AUC metric), data efficiency (to focus on the test-rig/run invariance of defect-detection methods, which has been the subject of very few research studies), and running time.

1. INTRODUCTION

Rotating machines are ubiquitous across all industrial sectors and they usually need to remain operational for an extended period and in harsh environments, which causes degradation and can eventually lead to failures in components. Bearings are one the most critical components in rotating machines. Bearing failures result in unplanned downtime and thus impact production and maintenance costs. To reduce costs, monitoring the condition of bearings, therefore, plays a vital role in the maintenance programs of all rotating machinery. It could allow to move from a traditional time-based preventive maintenance program to a condition-based maintenance or predictive maintenance strategy. An essential aspect in bearing health monitoring are methodologies that compute and/or calculate condition indicators (features) from noisy vibration signals acquired during early fault stages.

Various statistical features of vibration (or acoustic) data in the time domain are described in literature for bearing fault diagnosis (Freitas et al., 2016; Pichler et al., 2020), including variance, root mean square, peak-to-peak, kurtosis, and log-energy entropy. The bearing-fault related signal can be further enhanced by applying signal processing techniques such as cepstrum editing (Ompusunggu & Bartic, 2016; Peeters et al., 2016), and minimum entropy deconvolution (Sawalhi et al., 2007) in order to remove the effect of the transfer path and to reconstruct the source signal of the bearing faults.

Physics-based features such as the envelope spectrum of vibration data (Ompusunggu et al., 2019; Freitas et al., 2016; Kim et al., 2020) have gained popularity amongst the bearing health monitoring community given their robustness in detecting bearing inner race faults, outer race faults, ball and cage faults. The diagnostic information is contained primarily in the repetition rate of the pulses arising from bearing fault impacts. While spectral correlation is able to find a suitable band for demodulation of the vibration signal to obtain the envelope spectrum in general, it is shown in (Randall et al., 2016) that in the case where the signal is impulsive, a method based on kurtogram provides better results.

Other methods have been described in literature to compute features for bearing fault diagnosis. For example, the discrete wavelet transform has been applied in (Nishat Toma & Kim, 2020) to decompose an induction motor current signal into several coefficients. The statistical features of these decomposed coefficients are subsequently the input to two ensemble machine learning models, namely random forest and extreme gradient boosting, where both classifiers showed promising results to distinguish between healthy bearings, and bearings with inner faults and outer faults. As another example, (Al-Bugharbee & Trendafilova, 2015) used fitted auto-regressive model coefficients as input features to a nearest neighbor machine learning algorithm, in order to classify four bearing

fault conditions. Finally, convolutional neural networks have been widely used for bearing fault classification. In particular, (Janssens et al., 2016) show that convolutional neural networks can outperform various statistical features.

In most industrial applications, bearing fault detection methods ought to: (1) be data-efficient: one is faced, in many industrial settings, with a lack of training data, a lack of ground truth and scarce information on machine design and operational conditions and (2) require relatively low running time during inference, as these methods are in most cases deployed in edge devices running on a battery. In light of these technological challenges, we apply wavelet scattering networks to bearing fault detection (Ambika, Rajendrakumar, & Ramchand, 2019; Heydarzadeh & Mohammadi, 2017). This technique is based on the wavelet transform, where it computes translation-invariant features that are locally stable to deformation, making it particularly useful for classification and clustering purposes. Moreover, we specifically focus on the problem of test-rig/run invariance of defect detection models, which is an issue that has been the subject of very few research studies.

The method is experimentally validated with data acquired during 70 accelerated lifetime runs of bearings on 7 setups of the Smart Maintenance Living Lab of Flanders Make (Ooijevaar et al., 2019). Furthermore, the wavelet scattering network feature extraction method is compared to: (1) various statistical features, (2) the squared envelope spectrum method, and (3) a detection method based on convolutional neural networks (CNNs). This benchmark is done on three industrially-relevant performance metrics, taking into account accuracy (both a industrially-motivated weighted accuracy metric and the classic AUC metric), data efficiency (focusing on test-rig/run invariance of defect detection models), and running time.

2. METHODOLOGY

This section discusses the proposed methodology comprising 1) the wavelet scattering networks for features generation, subsequently followed by 2) principal component analysis (PCA) for dimensional reduction and 3) logistic regression for health assessment and classification.

2.1. Wavelet scattering networks

One of the difficulties in vibration-based bearing-fault classification stems from considerable variability within the classes themselves. For instance, bearing impact force variations during operation, manufacturing variability of mechanical parts, and changes in operating and environmental conditions can induce various deformations and rigid translations, which can influence amplitudes, phases, and frequencies of the data.

The wavelet scattering network (Bruna & Mallat, 2013) is a feature-generation method based on the wavelet transform. The obtained features are translation invariant and stable to small deformation, which increases the discrimination between classes, while still being continuous and stable to small variations. This has been extensively described in (Bruna & Mallat, 2013) using image data from the MNIST dataset, and can be summed up by the following Lipschitz continuity equation:

$$\|S(L_\tau x) - S(x)\| \leq \alpha \|x\| \|\nabla \tau\|_\infty, \tag{1}$$

where:

- S is the scattering wavelet operator,
- $L_\tau x(u) = x(u - \tau(u))$ is the translation and deformation operator,
- α is a constant, and
- $\nabla \tau$ is the deformation gradient tensor whose norm measures the deformation amplitude.

When the deformation operator L_τ is only translating the input x, i.e., $x(u - c)$, where c is a constant, this means that $\nabla \tau(u) = 0$, yielding translation invariance of the wavelet scattering transform. If the deformation operator L_τ is more complex, Eq. (1) shows that the deformed signal and the signal itself have a scattering distance essentially proportional to the deformation amplitude.

The use of the wavelet scattering transform has been described in the literature on several use cases such as audio classification (Andén & Mallat, 2011), where features have been computed from audio data and classified using an affine model selection using PCA. WSN features have also been applied in medical applications, where, e.g., (Liu et al., 2020) used the extracted features from ECG signals to classify four types of arrhythmia using neural networks trained on a reduced WSN feature space together with K-nearest-neighbors. The quality of scattered feature information has also been shown in (Wang et al., 2018), where it has been used as part of an algorithm of synthetic aperture radar (SAR) for automatic target recognition.

The basic building block of the WSN is Morlet wavelet. A dictionary of Morlet wavelets, see Eq. (2), is obtained by scaling and translating (with the operator T_τ) a single band-pass filter:

$$\psi_{j,\tau}(x) = 2^{-2j} \psi(2^{-j} T_\tau x) \tag{2}$$

At the zeroth order, a scaled Gaussian (Eq. (3)) is applied to the modulus of the input signal. This results in the first coefficients (Eq. (4)); they can be seen as features representing energy information of the signal.

$$\phi_{2^J}(x) = 2^{-2J}\phi(2^{-J}x) \tag{3}$$

$$S_{0,J}x = |x| * \phi_{2^J} \tag{4}$$

At the first order, the input signal is filtered with Morlet wavelets $(\psi_{j,\tau})_{j,\tau}$ and then high frequencies are eliminated by the convolution with the scaled Gaussian ϕ_{2^J}. This results in the first-order coefficients, see Eq. (5):

$$S_{1,J}x = |x * \psi_{j_1,\tau_1}| * \phi_{2^J} \tag{5}$$

The high frequencies that were eliminated in the first order are recovered in the second order (Eq. (6)) using wavelets at a finer scale. This is done by convolutions of the filtered signals $|x * \psi_{j_1,\tau_1}|$ with wavelets at scales $2^{j_2} < 2^J$. Their modulus is scaled afterwards with the Gaussian filter. The resulting coefficients are called *scattered coefficients* as they are computed from the interference of the input signal x with two wavelets.

$$S_{2,J}x = ||x * \psi_{j_1,\tau_1}| * \psi_{j_2,\tau_2}| * \phi_{2^J} \tag{6}$$

Higher-order scattered coefficients are computed in a similar fashion. In this way, scattering wavelet transform "spreads" the energy of the signal x across the scattering coefficients of each order. This energy decays quickly as the order increases. In fact, it is shown in (Bruna & Mallat, 2011) that 98% of the energy of scattered coefficients is carried by the zeroth, first, and second orders. Therefore, throughout the rest of the paper, we limit ourselves to these orders.

The final scattering matrix concatenates scattering coefficients of all orders to represent the features of the input signal, see Eq. (7):

$$S_{\text{final}}x = \{S_{m,J}\}_{m\in\{0,1,2\}} \tag{7}$$

The structure of wavelet scattering networks is similar to that of convolutional neural networks, but with different properties:

- The filters used to extract frequency information are not learned but are predefined Morlet wavelets.
- The modulus operator can be interpreted as a pooling function in the context of convolutional networks as it recombines real and imaginary parts of the coefficients.
- The scaled Gaussian averaging filter ϕ_{2^J} is also a pooling operator as it down-samples the input signal.

In the rest of the paper, the aggregated wavelet scattering transform features will need to be compressed before using them as input to traditional machine learning algorithms. This will be substantiated in Sections 2.2 and 2.3.

2.2. PCA

In order to avoid the dimensionality curse when classifying the generated data from the wavelet scattering transform, the latter is projected onto an orthonormal basis of lower dimension. This basis is found using principal component analysis (PCA).

The principal components are computed using the singular value decomposition of the aggregated wavelet scattering transform data matrix $\mathbf{X}_{\text{WSN}}$ (see Eq. (8)). The matrix $\mathbf{V}$ is used to map the data from the original feature space to the reduced one.

$$\mathbf{X}_{\text{WSN}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*, \tag{8}$$

where:

- $\mathbf{U}$ are the orthonormal left singular vectors,
- $\mathbf{\Sigma}$ is the diagonal singular values matrix, and
- $\mathbf{V}$ are the orthonormal right singular vectors.

2.3. Logistic Regression

In general, feature values are not necessarily restricted between 0 and 1, which does not allow a direct justification on the health of a Component Under Test (CUT). Despite reflecting the actual condition of a CUT, principal features extracted from measurement data cannot be directly used to assess the health of the CUT unless the *relative distances* to the corresponding values which represent the end of life of the CUT (*i.e.*, thresholds) are known. To this end, the feature values evolving from a healthy to failure state need to be transformed to health indices.

In this paper, we use logistic regression to transform feature values into health indicator values. As will be shown later, logistic regression can be seen as a process with a twofold objective: (i) fusing multiple features (independent variables) into a single value, the health index, and (ii) restricting the health index between 0 and 1, with 0 considered healthy and 1 faulty. As discussed in (Lemeshow & Hosmer, 2000; Kleinbaum et al., 2002), logistic regression is an appropriate technique for dichotomous problems, where the predicted variable (in this case the health index) must be greater than or equal to zero and less than or equal to one. Unlike linear regression which is inappropriate for dichotomous problems (Lemeshow & Hosmer, 2000; Kleinbaum et al., 2002), in logistic regression, only data representing healthy and failure states are required to estimate the regression coefficients. Thus, a logistic-regression technique is suitable for problems with limited history data (Ompusunggu et al., 2012).

Let us consider a simple **logistic function** $P(F)$ defined as:

$$P(\mathbf{F}) = h = \frac{1}{1+e^{-g(\mathbf{F})}} = \frac{e^{g(\mathbf{F})}}{1+e^{g(\mathbf{F})}}, \tag{9}$$

where $\mathbf{F} = \{F_1, F_2, \ldots, F_L\}$ denotes a set of L extracted features, h denotes the health index of an event (*i.e.* healthy or failure) given a set of features $\mathbf{F}$ and $g(\mathbf{F})$ is the **logit function** which is mathematically expressed as:

$$g(\mathbf{F}) = g = \log\left(\frac{P(\mathbf{F})}{1-P(\mathbf{F})}\right) = \sum_{i=0}^{L} \beta_i F_i, \tag{10}$$

where $F_0 = 1$, β_i denotes the logistic model parameters to be identified, and g denotes the logarithm of the "odds-of-success". In a more compact way, Eq. (10) can be rewritten as:

$$g = \boldsymbol{\beta}^{\mathrm{T}}\mathbf{F}, \tag{11}$$

with

$$\boldsymbol{\beta} = [\beta_0 \quad \beta_1 \quad \beta_2 \quad \cdots \quad \beta_L]^{\mathrm{T}}$$

and

$$\mathbf{F} = [1 \quad F_1 \quad F_2 \quad \cdots \quad F_L]^{\mathrm{T}},$$

where the superscript T denotes a transpose operation.

Note that the logistic function expressed in Eq. (9) can be seen as a kind of probability function (cumulative distribution function) because it ranges between 0 (healthy) and 1 (failure). In addition to this, the logit function expressed in Eq. (11) constitutes a linear combination of features extracted from measurement data $F_1, F_2, \ldots, F_L$. This implies that the logarithm of the odds-of-success g preserves the nature of features to be extracted from measurement signals.

Here, the main objective of the logistic regression is to identify the $L+1$ parameters of $\boldsymbol{\beta}$ in Eq. (11) such that the logistic model is readily implementable for the health assessment of a CUT. In this context, the parameter identification is normally performed using the maximum-likelihood estimator, which entails finding the set of parameters for which the probability of the observed data is maximal (Czepiel, n.d.). This is done off-line, where two sets of features, F_{healthy} and F_{failure} representing healthy and failure states, respectively, are used as training data.

3. Experimental validation

In this section, the wavelet scattering transform is used to extract features from acquired data and a comparison is made to other methods, namely statistical features, the squared envelope spectrum as well as a convolutional neural network based-approach. This benchmark will focus on three industrially relevant performance metrics taking into account accuracy, data efficiency, and running time.

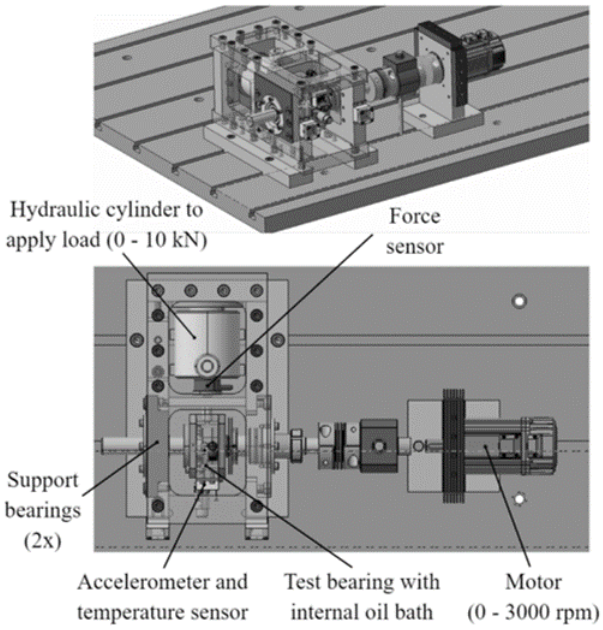

Figure 1. One of the seven test rigs.

3.1. Datasets

The Smart Maintenance Living Lab (Ooijevaar et al., 2019) consists of seven identical drive train sub-systems representing a fleet of machines, on which accelerated life time tests of bearings have been performed. During these tests reliable datasets of degrading bearings have been acquired.

One of these seven identical experimental rigs is depicted in Figure 1. It comprises of a single shaft with the test bearing that is supported on each side by a support bearing. A hydraulic cylinder applies a radial load of $F = 9$ kN to the bearing. The bearing is lubricated by an internal oil bath. The set-up is driven by a motor at a rotation speed of 2000 rpm. The accelerometer and load and speed sensors are mounted on the experimental set-up. The acquisition of the data is done through an industrial Beckhoff platform. Seventy runs have overall been gathered: per rig seven runs to end of life (defined as a peak to peak of 20 g) starting with a small initial indent with the average diameter of 400 micron on the bearing inner race, and three healthy bearing runs. The average length of a run is several hours. Vibration data is acquired at a sampling frequency of $F_s = 50$ kHz.

3.2. Results

After applying the scattering transform on the input signals from the dataset (by slicing the datapoints into 1s segments), a principal component analysis (PCA) has been applied to the resulting features to reduce the dimension of the space, while preserving as much variability between the data points as possible. Figure 2 illustrates the distribution of the datasets in a

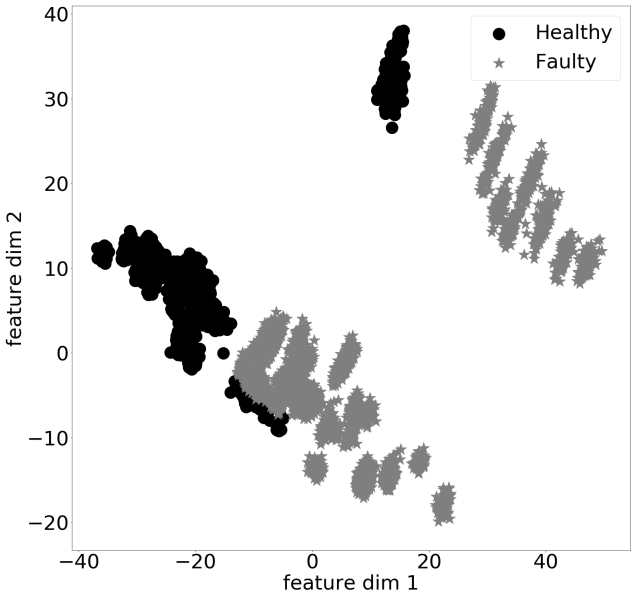

Figure 2. Distribution of wavelet scattering network features along the first two principal components.

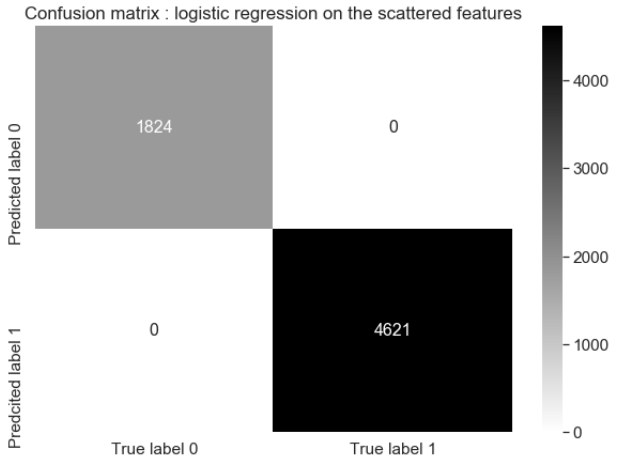

Figure 3. Confusion matrix of a classifier (logistic regression) trained on wavelet scattering network features (using a random split of 10% training data and 90% test data).

two-dimensional space obtained by PCA. It shows a separation between the two classes, i.e., datasets of the faulty and healthy state. From Figure 2 we observe the following:

- Some healthy data points are very close (distance-wise) in the reduced feature space to the faulty datapoints. After investigating these datapoints, it has been observed that they all belong to the same run of a bearing test. These datapoints exhibit anomalies relative to other healthy datasets especially in terms of amplitude (the reason for this is not clear and has not been investigated further yet). However, during performance evaluation in this paper, all data points were accounted for.
- Clustering of data points within each class can be clearly seen in Figure 2, within the same class; bearing signal coming from the same run/dataset exhibit higher similarities than data points coming from two different runs.
- Two clusters form within each class. This behavior has not been further examined yet.

A supervised model can then be trained on the compressed feature space to classify datasets of the faulty and healthy state. The number of PCA components to be kept is a tuning parameter during the training and testing of the classifier. Two PCA components are used in Figure 2 for illustration purposes only, as it has been observed after tuning that seven PCA components is most suitable for classification.

The confusion matrix (describing the true/false positives/negatives) of the linear classifier obtained from using logistic regression (Kleinbaum et al., 2002) on the feature values of the 70 datasets is shown in Figure 3. The classifier model is trained randomly on a subset of the datasets (a fraction of data points are taken from each of the 70 datasets available). Seven principle components are used. It can be seen that the model is able to successfully separate the two classes in a linear fashion. The fact that the training-testing split was done randomly dramatically increases the probability of having a perfect score, because the model will be seeing data points coming from different runs. In general and as can be seen from Figure 2, if only a few setups/runs are considered during the training, then the model will find the optimized separation while ignoring all the dataset coming from other, not considered runs/setups, and hence it will result in a worse score. The reason for this is, within the same class, there are generally more similarities between datapoints coming from the same accelerated life time test/run than datapoints coming from different tests/runs, this can be due to the way the test rig is commissioned for a certain run i.e. mechanical part assembly variations such as shaft misalignment and bearing preload variations. This challenge is further examined in Section 4.3 and a metric related to data efficiency will be introduced to evaluate test rig/machine invariance.

4. BENCHMARKING

4.1. Penalized accuracy metric

The penalized accuracy metric considered in this paper is the following:

$$\mathrm{Acc}_{\mathrm{penalized}} = 1 - (200 \cdot \mathrm{FPR} + \mathrm{FNR}), \qquad (12)$$

where FPR is the false positive rate and FNR is the false negative rate. The rationale behind this formula is that even though a false negative is considered about 5 times more severe (this factor is obtained from feedback from industry), in industrial applications, we can expect about three orders of magnitude more measurements of healthy datapoints, leading to

a weighting factor of $1000/5 = 200$ for FPR. Consequently, minimizing the false positive rate is imperative.

Note that perfect classification (i.e., FPR $=$ FNR $= 0$) corresponds to accuracy one, while with no skill it is possible to obtain accuracy zero by classifying every datapoint as healthy. Note however that accuracy can be negative. Accuracy is computed on models trained using a leave-one-dataset-out cross validation procedure, i.e., one dataset is tested against a model that is trained on the remaining datasets.

4.2. Area under the ROC curve

We also consider in this paper a normalized (in contrast with the one mentioned in Section 4.1) accuracy metric classically used to assess the performance of binary classifiers: the Area Under the ROC (Receiver Operating Characteristic (Fawcett, 2006)) Curve (AUC). We refer to (Fawcett, 2006) for a detailed discussion of the ROC curve and the AUC.

The AUC metric is chosen because, on the one hand, it is a normalized value to assess the performance of binary classifiers, and, on the other hand, it is a threshold-invariant metric that summarizes the trade-off between the true positive rate and false positive rate.

4.3. Data efficiency metric

In order to insure a robust diagnosis of bearing faults, it is important for defect detection models to be setup invariant. Moreover, in industry, often few datasets are available and in particular few faulty datasets. Therefore, it is important for defect detection models to perform well with little training data. In order to quantify this, we present in this section a novel metric.

Given a certain model, the data efficiency is evaluated in the following manner using a matrix, see Figure 4:

- Each value in this data efficiency matrix is computed by randomly selecting fixed numbers of healthy and faulty datasets on which the model is trained, and the model is then tested on the other datasets. This is iterated 50 times to cover multiple combinations.
- The accuracy values in the data efficiency metric are computed according to Eq. (12).
- In Figure 4, the maximum number of faulty datasets is purposely chosen less than the maximum number of healthy datasets to mimic industrial settings, where most often more healthy datasets are available than faulty datasets.

The data efficiency metric is defined as the "volume under the surface" of the data efficiency matrix, where we ignore negative accuracy values (since accuracy zero can already be accomplished by replacing it by a no-skill algorithm). More precisely,

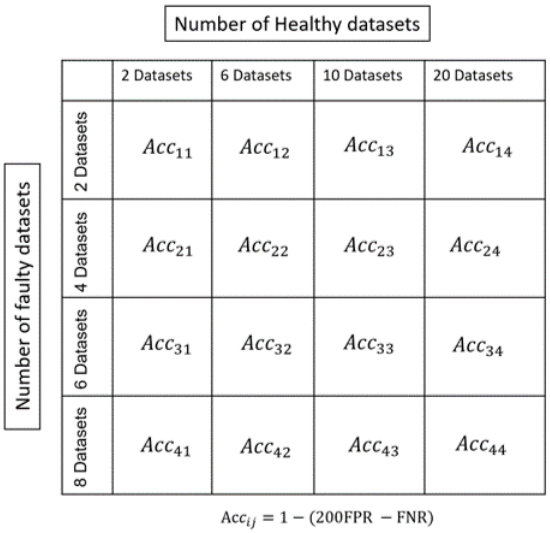

Figure 4. Data efficiency performance matrix for a classification algorithm.

$$\text{Data efficiency} = \frac{1}{|F| \cdot |H|} \sum_{f \in F} \sum_{h \in H} \max(\text{Acc}_{f,h}, 0), \quad (13)$$

where F is the set of numbers of faulty datasets (the row indices of the matrix of Figure 4) and H is the set of numbers of healthy datasets (the column indices of the matrix of Figure 4). In other words, data efficiency is the mean of the entries of the matrix obtained from Acc by replacing every negative entry by zero.

4.4. Comparison

In this section, five bearing fault features are compared to the wavelet scattering network algorithm. All computations were done on a laptop with an Intel Core i7-9850H CPU and 16 GB of RAM memory running Windows 10.

- Statistical features:
  - RMS: Root mean square (Eq. (14)) is the result after point-wise squaring, followed by taking the mean, followed by applying the root function. RMS is a measure for the amount of power dissipation.

$$\text{RMS}(x) = \sqrt{\frac{1}{n} \sum_{k=1}^{n} x_k^2}, \quad (14)$$

    where n is the length of x.
  - Peak: Peak is the highest value among the samples

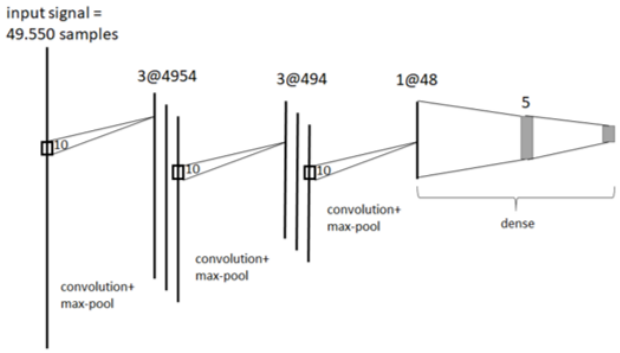

Figure 5. 1D CNN architecture

of the vibration signal (Eq. (15)).

$$\text{Peak}(x) = \max(x) \tag{15}$$

  - Kurtosis: Comparing the signal kurtosis to 3 gives a measure on the non-Gaussian nature of the signal.
- Physics-based features:
  - Squared envelope spectrum: the algorithm used in this paper is presented in (Freitas et al., 2016; Ompusunggu et al., 2019).
- AI-based features:
  - Applying a one-dimensional convolutional neural network (1D CNN) to the raw input: The network topology is given in Figure 5 and is similar to the network topology used in (Eren, Ince, & Kiranyaz, 2019) (in particular, the networks operate on raw vibration data and consist of three 1D convolutional layers with subsampling followed by dense layers). The first convolutional layer comprises of 3 filters that map the input into three vectors of length 4954, the second convolutional layer comprises of 3 filters that maps its input to three vectors of length 494, the third convolutional layer comprises of 1 filter that maps its input to one vector of length 48, the fourth layer consists of a dense layer that maps its input to one vector of length 5, then a classification layer that aims to differentiate between healthy and faulty states. The network topology and the used learning rate (of 10^{-4}) and momentum (of 0.9) of stochastic gradient descend have been manually optimized for accuracy.

Table 1 shows a comparison between different methods. It is clear that the logistic regression trained with the wavelet scattering network features along with the 1D CNN outperforms the other methods in terms of penalized accuracy and area under the ROC curve. However it can be shown that the running time of the wavelet scattering feature computation is one order of magnitude higher than the physics-based model, and two order of magnitudes higher than the CNN-based model, the latter is not practically an issue for most industrial applications as the features do not have to be computed continuously, but can be computed every other period of time (for example, every second or every minute). Nevertheless, the wavelet scattering feature method seems to represent the input signals in the feature space well as its data efficiency value is relatively high.

Table 1. Benchmark of various features according to the penalized accuracy, data efficiency, and running time metrics, respectively.

Method	Acc.	AUC	Data eff.	Running time (ms)
RMS	0.197	0.51	0.0036	0.074
Peak	0.182	0.47	0.019	0.022
Kurtosis	0.409	0.59	0.0525	1.3
SES	0.434	0.64	0.1942	17.0
CNN	0.593	0.97	0.0094	3.0
WSN	0.65	0.99	0.2336	600

5. Conclusion

This paper aims to show the performance and quality of wavelet scattering networks features for bearing defect detection compared to other methods: statistical features, namely, root mean square, peak, and kurtosis, physics-based features, namely the squared envelope spectrum, and a CNN-based fault detection method. The comparison was made using three metrics: (1) a penalized accuracy, where the goal is to minimize as much as possible false alarms while still being able to accurately detect defects, (2) data efficiency, in order to provide a metric to evaluate the machine/setup invariance of detection models, and (3) running time of detection models. It has been shown that the wavelet scattering networks features offers the highest accuracy and data efficiency relative to the other methods, but requires more time to compute. An important topic for future work is to compare the methods against datasets acquired using more involved drive trains (e.g., with gears) and under varying operating conditions.

Acknowledgment

This research received funding from the Flemish Government under the “Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen” program. This research was supported by Flanders Make, the strategic research centre for the manufacturing industry. We thank the reviewers for their helpful comments that have improved the paper.

References

Al-Bugharbee, H., & Trendafilova, I. (2015). Autoregressive modelling for rolling element bearing fault diagnosis. *Journal of Physics: Conference Series*, *628*(1), 2616-2633.

Ambika, P., Rajendrakumar, P., & Ramchand, R. (2019).

Vibration signal based condition monitoring of mechanical equipment with scattering transform. *Journal of Mechanical Science and Technology*, *33*(7), 3095-3103.

Andén, J., & Mallat, S. (2011). Multiscale scattering for audio classification. *In ISMIR*, 657-662.

Bruna, J., & Mallat, S. (2011). Classification with scattering operators. *In CVPR 2011*, 1561-1566.

Bruna, J., & Mallat, S. (2013). Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, *35*(8), 1872-1886.

Czepiel, S. (n.d.). Maximum likelihood estimation of logistic regression models: theory and implementation. *Available at* `czep.net/stat/mlelr.pdf`.

Eren, L., Ince, T., & Kiranyaz, S. (2019). A generic intelligent bearing fault diagnosis system using compact adaptive 1D CNN classifier. *Journal of Signal Processing Systems*, *91*, 179-189. doi: 10.1007/s11265-018-1378-3

Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, *27*(8), 861-874.

Freitas, C., Cuenca, J., Morais, P., Ompusunggu, A., Sarrazin, M., & Janssens, K. (2016). Comparison of vibration and acoustic measurements for detection of bearing defects. In *International conference on noise and vibration engineering 2016.*

Heydarzadeh, M., & Mohammadi, A. (2017). A robust feature extraction for automatic fault diagnosis of rolling bearings using vibration signals. In *In asme 2017 dynamic systems and control conference.*

Janssens, O., Slavkovikj, V., Vervisch, B., Stockman, K., Loccufier, M., Verstockt, S., ... Van Hoecke, S. (2016). Convolutional neural network based fault detection for rotating machinery. *Journal of Sound and Vibration*, *377*, 331-345.

Kim, S., An, D., & Choi, J. (2020). Diagnostics 101: A tutorial for fault diagnostics of rolling element bearing using envelope analysis in matlab. *Applied Sciences*, *10*(20), 7302.

Kleinbaum, D., Dietz, K., Gail, M., Klein, M., & Klein, M. (Eds.). (2002). *Logistic regression*. New York: Springer-Verlag.

Lemeshow, D., & Hosmer, S. (2000). *Applied logistic regression*. New York: Willey. ISBN 0-471-35632-8.

Liu, Z., Yao, G., Zhang, Q., Zhang, J., & Zeng, X. (2020). Wavelet scattering transform for ecg beat classification. *Computational and Mathematical Methods in Medicine*.

Nishat Toma, R., & Kim, J. (2020). Bearing fault classification of induction motors using discrete wavelet transform and ensemble machine learning algorithms. *Applied Sciences*, *10*(15), 5251.

Ompusunggu, A. P., & Bartic, T. A. (2016). Automated cepstral editing procedure (acep) for removing discrete components from vibration signals. *International Journal of Condition Monitoring*, *6*(3), 56-61.

Ompusunggu, A. P., Ooijevaar, T., Kilundu Y'Ebondo, B., & Devos, S. (2019). Automated bearing fault diagnostics with cost-effective vibration sensor. In J. Mathew, C. Lim, L. Ma, D. Sands, M. E. Cholette, & P. Borghesani (Eds.), *Asset intelligence through integration and interoperability and contemporary vibration engineering technologies* (pp. 463–472). Cham: Springer International Publishing.

Ompusunggu, A. P., Vandenplas, S., Sas, P., & Van Brussel, H. (2012). Health assessment and prognostics of automotive clutches. *PHM Society European Conference*, *1*(1).

Ooijevaar, T., Pichler, K., Di, Y., Devos, S., Volckaert, B., Van Hoecke, S., & Hesch, C. (2019). Smart machine maintenance enabled by a condition monitoring living lab. *IFAC-PapersOnLine*, *52*(15), 376-381.

Peeters, C., Guillaume, P., & Helsen, J. (2016). Signal preprocessing using cepstral editing for vibration-based bearing fault detection. In *In proc. int. conf. noise vib. eng.(isma)* (p. 2489-2502).

Pichler, K., Ooijevaar, T., C., Hesch, Kastl, C., & Hammer, F. (2020). Data-driven vibration-based bearing fault diagnosis using non-steady-state training data. *Journal of Sensors and Sensor Systems*, *9*(1), 143-155.

Randall, R., Antoni, J., & Gryllias, K. (2016). Alternatives to kurtosis as an indicator of rolling element bearing faults. In *In proceedings of isma2016 international conference on noise and vibration engineering* (p. 2503-2516).

Sawalhi, N., Randall, R., & Endo. (2007). The enhancement of fault detection and diagnosis in rolling element bearings using minimum entropy deconvolution combined with spectral kurtosis. *Mechanical Systems and Signal Processing*, *21*(6), 2616-2633.

Wang, H., Li, S., Zhou, Y., & Chen, S. (2018). Sar automatic target recognition using a roto-translational invariant wavelet-scattering convolution network. *Remote Sensing*, *10*(4), 501.