Interpretation of Deep Learning Models in Bearing Fault Diagnosis

Menno Liefstingh[1,2], Cees Taal[1], Sebastián Echeverri Restrepo[1,3] and Alireza Azarfar[1*]

[1] *SKF Research & Technology Development, Meidoornkade 14, 3992AE Houten, the Netherlands.*
menno@liefstingh.nl
cees.taal@skf.com
sebastian.echeverri.restrepo@skf.com
alireza.azarfar@skf.com

[2] *Faculty of Science and Engineering, University of Groningen, Nijenborgh 9, 9747 AG Groningen, the Netherlands.*

[3] *Department of Physics, King's College London, Strand, London WC2R 2LS, United Kingdom.*

ABSTRACT

In recent years, data-driven techniques such as deep learning (DL) have been widely represented in the literature in the field of bearing vibration condition monitoring. While these approaches achieve excellent performance in classifying bearing faults on controlled laboratory data sets, there is little information available about their applicability to more realistic working conditions. As a first step towards revealing the generalizability of DL models, we aim to understand the underlying representations that DL networks use to classify bearing defects. An interpretable DL model can give us hints on how to increase its transferability by, e.g., using data augmentation, changing input representations and/or adapting model architectures. We use the Grad-CAM algorithm along with signal transformations to identify the elements of the input spectrogram that contribute to class attribution. The results show that removing time-domain information from the spectrogram has a minor impact on its performance. Instead, the network learns distinct average frequency profiles. We therefore conclude that the networks learn signal features very specific to the physical properties of the specific test setup, such as the frequency response function, rather than more general features related to bearing defects.

1. INTRODUCTION

Rolling element bearings are widely used in a plethora of rotating equipment and are critical components for their adequate performance. Bearing failures can lead to unplanned downtime with unforeseen costs, or even result in potential disasters. Sensor-based condition monitoring has been an important tool for the prediction of these undesired events. Using vibration sensors to monitor the condition of a bearing is a common practice in industry and has been a well studied topic in academia (see the work from Randall and Antoni (2011) for an overview).

Traditionally, bearing fault diagnosis has been based on physics-inspired signal processing techniques, where time-frequency analysis methods are applied to analyze vibration signals. Such analyzes have been used to reveal the location of surface defects (e.g., a spall on the inner or outer race) (Randall & Antoni, 2011), and to estimate the size of such faults (Epps, 1991; H. Zhang et al., 2021). Knowledge of the spall sizes can be used as a tangible measure of the severity of a fault and can therefore be of substantial help for the optimization of maintenance intervals.

One of the main challenges for the detection and size estimation of spalls in real applications, is that the changes of the vibration signatures (which are a function of spall sizes) are difficult to observe due to their low signal to noise ratios (SNR). Furthermore, the majority of the information available in the literature on the topic of spall size estimation is based on controlled setups with very clean signals (H. Zhang et al., 2021; Epps, 1991). Although these studies have made important contributions to the understanding of the dynamics of bearing containing systems, the algorithms proposed are difficult to generalize to real applications.

Deep learning (DL) has introduced many breakthroughs in the fields of computer vision, speech recognition and natural language processing (LeCun, Bengio, & Hinton, 2015) and there is currently great interest to study its potential in the field of bearing fault diagnosis. In the last decade, a number of studies have been published where data-driven approaches, such as DL, are used in the field of bearing fault diagnosis

*Corresponding author.

(see S. Zhang, Zhang, Wang, and Habetler (2020) for an extensive review). In contrast to traditional signal processing approaches, DL does not depend on human-engineered features, but automatically learns abstract signal representations to distinguish bearing fault classes.

Despite all the recent progress in DL, there are still many challenges that need to be overcome in order to apply DL on more realistic working conditions (S. Zhang et al., 2020). For instance, due to the lack of labeled data from real applications, DL methods are typically trained on labeled laboratory data sets such as the one from the Case Western Reserve University (CWRU) (Smith & Randall, 2015). However, little is known about the ability of these models to transfer their "knowledge" from the laboratory to the real world.

To improve the generalizability of DL methods for bearing health state diagnosis, different approaches have been proposed in the literature. One common technique is to apply transfer learning methodologies, such as domain adaptation (DA) (Ganin & Lempitsky, 2014), to force the network to learn similar feature representations between different domains. For example, in Wang, Michau, and Fink (2019) torque loads were used to define the different domains. In C. Liu, Mauricio, Qi, Peng, and Gryllias (2020), fault vibration signals were generated with a physical model and DA was used to close the gap between the real and synthetic vibration signals.

Another approach is to introduce domain knowledge by preprocessing the signal into a more meaningful input representation, allowing the network to learn more general features. For example, in Chen, Mauricio, Li, and Gryllias (2020) a 2D cyclic spectral coherence representation was used to exploit the second-order cyclostationary behavior of bearing vibration signals (Antoni, 2009). Although the aforementioned approaches are a good step towards more generalizable DL methods, their validation has been limited to laboratory data sets. Their applicability to real world data still remains to be demonstrated.

In the present article we investigate what kind of signal features typical DL methods (e.g., (S. Zhang et al., 2020)) actually learn, and investigate their potential to estimate spall sizes in a more general way. Having an interpretable DL model can give us hints on how to increase its applicability by, e.g., using data augmentation, changing input representations and/or adapting model architectures. To this end, we analyze a typical setup from the DL methods in S. Zhang et al. (2020). We evaulate two different convolutional neural networks (CNNs) trained on input spectrograms to classify bearing faults from two different datasets. We use the Grad-CAM algorithm (Selvaraju et al., 2019) together with signal modifications to evaluate which parts of the input signal contribute to class attribution.

2. METHODS

The interpretable deep learning framework is illustrated in figure 1 and is described by the following steps: First, the vibration signals are pre-processed to obtain a spectrogram representation. Second, the input representations are fed into two different CNN architectures, one with a general baseline architecture and one state-of-the-art network. Third, the Grad-CAM procedure is used to generate activation maps that reveal the regions of importance for classification on the input representations. Finally, to supplement the Grad-CAM activation maps we propose several signal processing methods to modify signals to change the classification performance and therefore validate the interpretation of the model.

2.1. Spectrogram input representations

Spectrograms are visual 2D representations of the frequencies of a signal as a function of time. In this work they are obtained by taking the logarithm of the squared magnitude of the short-time Fourier transform (STFT) of the signal. We transform the raw signals into spectrograms for the following two reasons. First, spectrograms resemble a typical input transformation used in DL methods from literature (Tao, Wang, Chen, Stojanovic, & Yang, 2020; Verstraete, Ferrada, Droguett, Meruane, & Modarres, 2017; H. Liu, Li, & Ma, 2016). And, second, spectrograms are image-like representations that permit the use of recent advancements in the field of computer vision model interpretability, such as the aforementioned Grad-CAM approach.

For the generation of the spectrograms, a signal duration of $500\,\mathrm{ms}$ is used, containing more than ten defect pulse repetitions. For the STFT settings, a Hanning window with a size of 256 samples and an overlap of 50% are used. This representation is then resampled to an image resolution of 112 by 112 pixels where the levels are normalized in such a way that the full dynamic range of the image is used. These settings are chosen to resemble typical setups available in the literature (Tao et al., 2020; Chen et al., 2020).

2.2. Network architectures

As previously mentioned, two neural network architectures (A and B) were considered. Network A has a rather general and simple architecture, representative of what is used in the literature. The architecture of network B is inspired by one of the state of the art networks available in the literature.

2.2.1. Network architecture A

Network A is a CNN designed with a relatively simple architecture to be able to rule out any effects that might arise from non-standard network elements, like group normalization in the approach by Chen et al. (2020). A simple network will also allow us to determine if the obtained results could be ex-

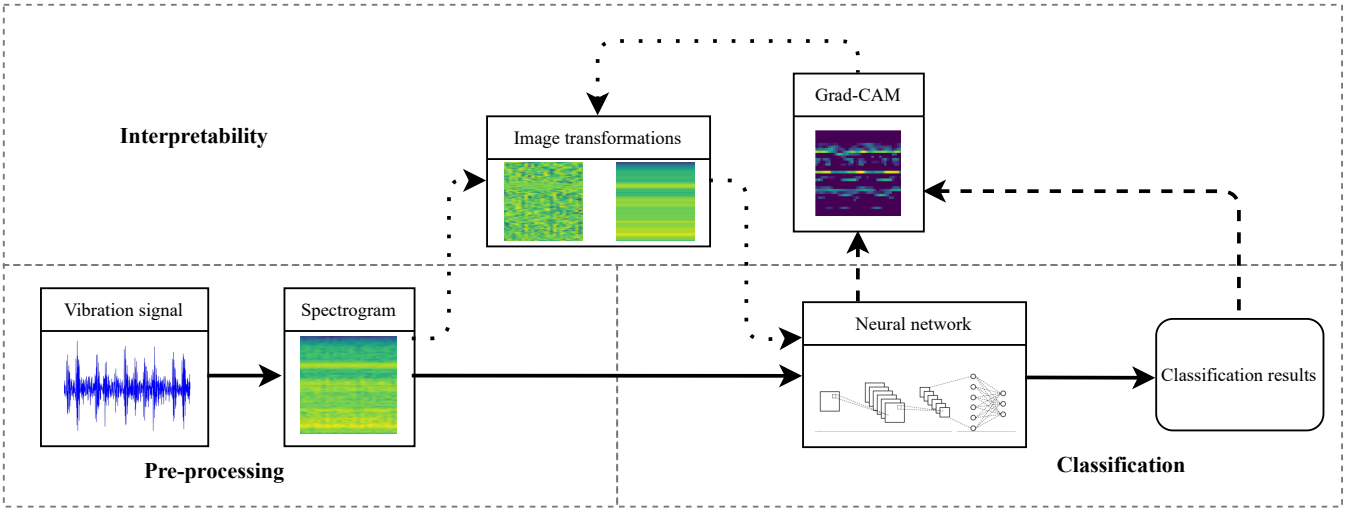


Figure 1. Flowchart of the methodology used for the interpretability study of bearing vibration signals

tended to other types of (convolutional) neural networks. The architecture of network A is presented in table 1.

After random initialization of the network weights, we train the network with a batch size of 32 for 50 epochs. Each epoch takes approximately 5 s when using GPU acceleration or 30 s without it.

2.2.2. Network architecture B

For Network B, we use a convolutional neural network inspired by the work of Chen et al. (2020). This network architecture employs multiple convolutional layers with group normalization layers for regularization. The selection of this specific network, is motivated by the fact that we want to gain an insight into how the state-of-the-art networks in the literature attain such high performance scores in fault classification tasks. A full overview of network B is presented in table 2.

The network is trained using a batch size of 32 for 20 epochs and randomly initialized weights. Each (GPU-accelerated) training epoch takes approximately 5 s, so training the network can be done in under two minutes. When training the network on a CPU, a single epoch takes approximately 30 s.

2.3. Interpretability

2.3.1. Grad-CAM

Grad-CAM is a method for attention visualization of convolutional neural networks originally introduced by Selvaraju et al. (2019). The Grad-CAM algorithm uses a trained neural network along with an input to highlight which areas of an input image are important for classification. With the spectrogram representation used in this research, we can use Grad-CAM to highlight which parts are important for classification.

We use the implementation of the original Grad-CAM algorithm by Chattopadhyay, Sarkar, Howlader, and Balasubramanian (2017) as made available on their GitHub page[1]. This implementation follows the original article that proposes Grad-CAM (Selvaraju et al., 2019).

The Grad-CAM algorithm obtains the class-discriminative localization map $L^c_{\text{Grad-CAM}} \in \mathbb{R}^{u \times v}$ of width u and height v for any class c by computing the gradient $\frac{\partial y^c}{\partial A^k}$ of the score for class c, denoted as y^c, with respect to the feature map activations A^k of a convolutional layer. These computed gradients are then global-average-pooled over the width (i) and height (j) dimensions to obtain the neuron importance weights α^c_k:

$$\alpha^c_k = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A^k_{ij}} \tag{1}$$

This weight α^c_k represents a *partial linearization* of the deep network downstream from A and therefore captures the importance of feature map k for a target class c. The algorithm then performs a weighted combination of forward activation maps followed by a ReLU to obtain:

$$L^c_{\text{Grad-CAM}} = \text{ReLU}\left(\sum_k a^c_k A^k\right) \tag{2}$$

This results in a heatmap with the same size as the convolutional feature map of the last convolutional layer (37×37 or 56×56 for the respective network architectures). The reasoning behind the application of a ReLU to the linear combination of maps is that Grad-CAM should only find the features that have a *positive* influence on the class of interest (Selvaraju et al., 2019). The resulting feature maps are plotted to a colormapped image for interpretation.

[1] https://github.com/samson6460/tf_keras_gradcamplusplus

Layer	**Layer type**	**Parameter**	**Setting**	Filter size	Padding	Output size
1	Input	/	/	/	/	(112, 112)
2	2D Conv	# of kernels	16	3×3	Yes	(112, 112, 16)
3	2D Conv	# of kernels	32	3×3	Yes	(112, 112, 32)
4	Dropout	Dropout rate	0.2	/	/	(112, 112, 32)
5	Max Pooling	/	/	3×3	/	(37, 37, 32)
6	2D Conv	# of kernels	32	3×3	Yes	(37, 37, 32)
7	Dropout	Dropout rate	0.2	/	/	(37, 37, 32)
8	Max Pooling	/	/	3×3	/	(12, 12, 32)
9	Softmax output	# of outputs	12	/	/	(12, 1)

Table 1. Convolutional neural network architecture A

Layer	**Layer type**	**Parameter**	**Setting**	Filter size	Padding	Output size
1	Input	/	/	/	/	(112, 112)
2	2D Conv	# of kernels	16	3×3	Yes	(112, 112, 16)
3	Group Norm	Group size	16	/	/	(112, 112, 16)
4	Max Pooling	/	/	2×2	/	(56, 56, 16)
5	2D Conv	# of kernels	32	3×3	Yes	(56, 56, 32)
6	Group Norm	Group size	16	/	/	(56, 56, 32)
7	Max Pooling	/	/	2×2	/	(28, 28, 32)
8	Fully Connected	# of nodes	256	/	/	(256, 1)
9	Group Norm	Group size	16	/	/	(256, 1)
10	Dropout	Dropout rate	0.2	/	/	(256, 1)
11	Fully Connected	# of nodes	126	/	/	(126, 1)
12	Group Norm	Group size	16	/	/	(126, 1)
13	Dropout	Dropout rate	0.2	/	/	(126, 1)
14	Softmax	# of outputs	12	/	/	(12, 1)

Table 2. Convolutional neural network architecture B

Additionally, we generate an "average Grad-CAM map" by letting the network predict all examples in the test set, summing the Grad-CAM arrays for each class and dividing them by the number of correct examples in each respective class. We chose to exclude misclassifications from these average Grad-CAM maps because the Grad-CAM algorithm takes the predicted class as input; including incorrect predictions would dilute the resulting average maps with representations of other classes.

2.3.2. Input transformations

To further interpret the models, two different transformations are applied to the spectrograms in order to isolate the impact of some features on the classification results. The goal is to determine whether the network is using/learning common generalizable signal features (already known from traditional vibration analysis), or other information that might not be related to the presence of defects/faults in the bearing. The signal transformations also help us to see if the features that contribute the most to the classification are in line with the Grad-CAM activation maps. In the results section, the relation between the Grad-CAM maps and the signal transformations will be further explained.

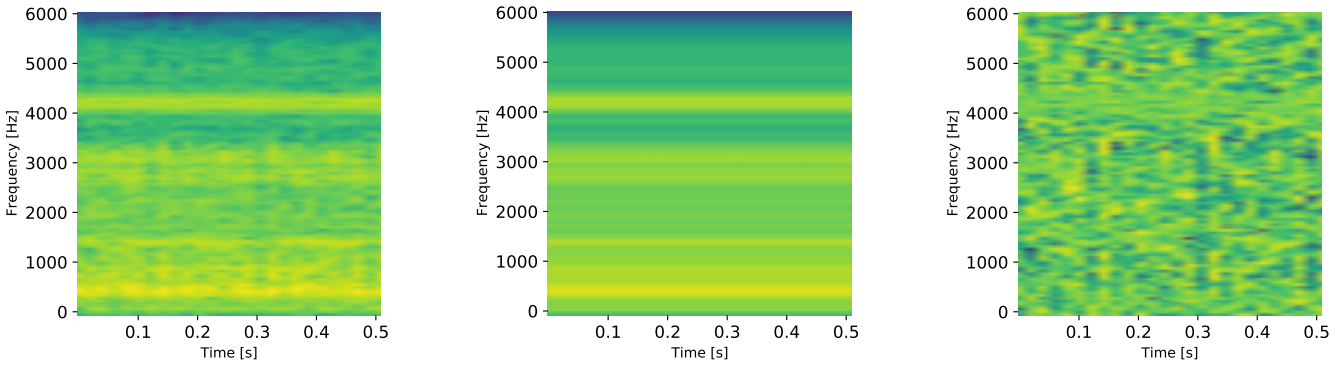
As will be clear from the experimental Grad-CAM results, we suspect that the network might be sensitive to average frequency profiles. These are related to the resonating frequencies in the transfer function of the specific test setup rather than conventional signal features. We therefore propose the following two signal transformations:

- **Time-information removal**: Each row in the spectrogram is replaced by its mean value in order to remove the time-domain structure. As a consequence, traditional signal information related to repetition frequencies is removed, while transfer function information is preserved.
- **Frequency normalization**: Each spectrogram row is normalized to unit energy before the log-transform is applied. This will eliminate the effect of the transfer function.

Examples of the transformations are shown in figure 2. Figure 2a shows the unedited spectrogram without any transformations applied to it. Figure 2b and 2c show the the time-information removal and frequency normalization transformations applied to them respectively.

3. RESULTS

Two network architectures are analyzed with two independent data sets to control for their effects on the learned representations. First, results of the analysis made using the CWRU data set are presented, followed by the outcome of the interpretability study using a second internal SKF data set.



(a) Input spectrogram. (b) Time-information removal. (c) Frequency normalization.

Figure 2. An overview of the input transformations applied to a single example. The color scale goes from blue (low) to yellow (high).

3.1. CWRU data set

The data set published by Case Western Reserve University (CWRU) contains ball bearing data for normal and faulty bearings mounted on a test bench powered by an electric motor. The vibration signals are recorded by several accelerometers mounted on multiple locations on the test setup, and accompanied by meta-data including size and location of the defect together with speed of rotation and torque load. The data and further experimental details are available online [2].

For our analysis the drive-end data is selected. It has a median length of approximately 10 s and it was recorded using a sample rate of 12 kHz. The data set is separated into 12 different classes (see the overview in table 3).

Defect location	Defect size	Class name
Ball	0.007 in	B007
Ball	0.014 in	B014
Ball	0.021 in	B021
Ball	0.028 in	B028
Inner race	0.007 in	IR007
Inner race	0.014 in	IR014
Inner race	0.021 in	IR021
Inner race	0.028 in	IR028
Outer race	0.007 in	OR007
Outer race	0.014 in	OR014
Outer race	0.021 in	OR021
-	-	Normal

Table 3. Overview of different CWRU classes

3.1.1. Performance of the networks

Our custom network architecture A attains an average test accuracy of 94.58%, with a standard deviation of 3.44%, in a ten-repetitions experiment using different train-test splits. For the Grad-CAM and input transformation experiments, we used a fixed train-test split to be able to compare the results. The classification accuracy of network architecture A on this fixed train-test set is 94.8%, which aligns well with the classification accuracy obtained on the ten-repetitions experiment. The F1-scores per class and network are presented in table 4. We see that, apart from the B014, B021 and OR14 classes, the classification performance of network architecture A is remarkable. This high performance indicates that, despite the low complexity of the network, it has learned and stored features that can clearly separate the different classes.

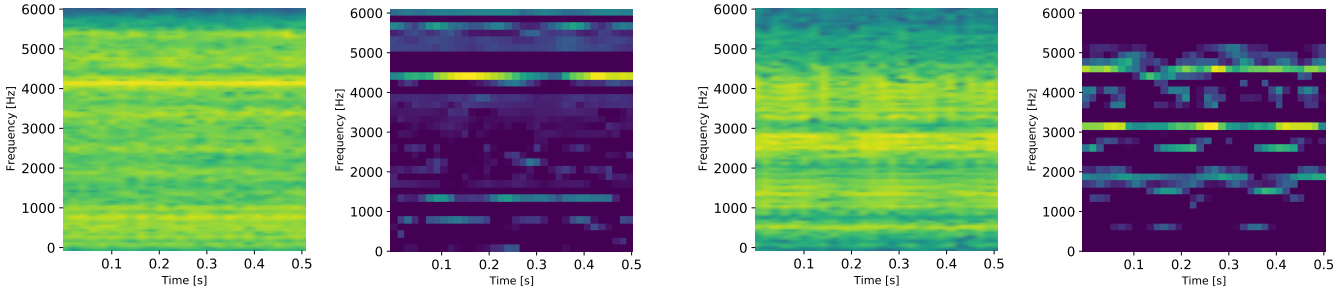
Network architecture B, using the same training schema, reached an even higher average classification accuracy of 97.7%, with a standard deviation of 3.91%. A classification accuracy of 96.9% was obtained using the fixed train/test split. The corresponding F1-scores per class are presented in table 4.

The improved performance compared to network architecture A can be explained by the higher complexity of this network architecture: network architecture A has only 69 644 trainable parameters compared to 6 463 180 trainable parameters in network architecture B. Alongside this significant increase in the number of training parameters, network architecture B benefits from group normalization layers that provide regularization which could potentially contribute to the higher classification accuracy.

3.1.2. Interpretability

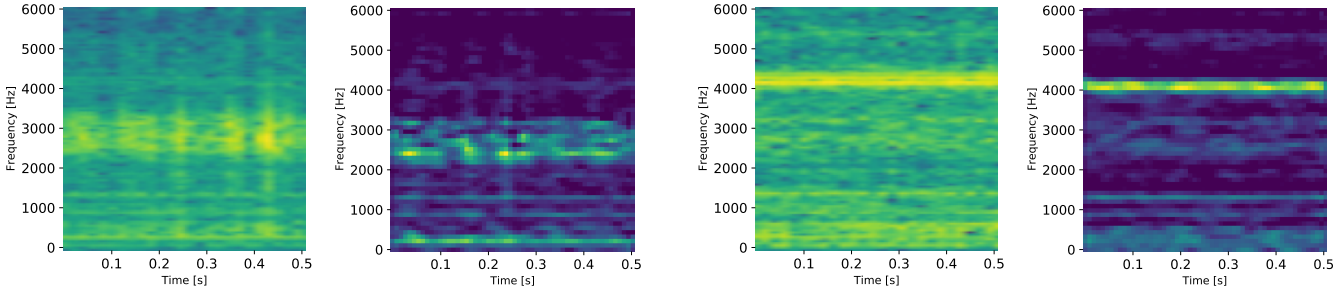
For each network type, two spectrograms with their corresponding Grad-CAM activation maps are shown in figures 3 and 4. The activation maps indicate the potential importance of the resonances of the system in the classification results. These resonances, characterized by high energy concentrations in distinguished frequency bands, are mainly parameters of the transfer function of the system, rather than being uniquely related to the bearing defect excitation signal. These resonances can be seen in figures 3 and 4 as horizontally dis-

[2] https://engineering.case.edu/bearingdatacenter



(a) Input image (B021) (b) Grad-CAM map (B021) (c) Input image (IR007) (d) Grad-CAM map (IR007)

Figure 3. Examples of Grad-CAM activation maps of network architecture A trained on the CWRU data set. The color scale goes from blue (low) to yellow (high).



(a) Input image (B014) (b) Grad-CAM map (B014) (c) Input image (normal) (d) Grad-CAM map (normal)

Figure 4. Examples of Grad-CAM activation maps of network architecture B trained on the CWRU data set. The color scale goes from blue (low) to yellow (high).

Class	Network A	Network B
B007	1.000	1.000
B014	0.876	0.962
B021	0.755	0.848
B028	1.000	1.000
IR007	1.000	1.000
IR014	0.960	0.927
IR021	1.000	1.000
IR028	1.000	1.000
OR007	1.000	1.000
OR014	0.788	0.892
OR021	1.000	1.000
Normal	1.000	1.000
All test samples	**0.948**	**0.969**

Table 4. Classification performance (F1-scores) of network architecture A and B for different classes

tributed higher energies (represented by yellow) at certain frequency bands.

Motivated by the insights given by the Grad-CAM activation maps it is of interest to see if the network is analyzing any temporal structure with respect to any of these horizontal lines. Therefore, we apply the transformations as discussed in Section 2.3.2.

The results presented in figure 5 show the effect of the input transformations on the classification performance using network architecture A. As clearly seen in the figure, removing the temporal information from the signal, where the resonances of the system are emphasized in the input image, has negligible effect on the classification performance. On the contrary, suppressing the transfer function effect in the signal by normalizing along the frequency axis, dramatically weakens the performance of the model. Only a few of the classes are partially correctly classified and, even those cases, are questionable. For instance, additional experiments showed that class "OR021" can be classified with high confidence by feeding a spectogram of random Gaussian noise to the network. Altogether, these experiments confirm the hypothesis made based on the Grad-CAM outcome, where it was suggested that the network picks up features related to resonances of the system rather than information related to the bearing defects.

Figure 6 shows the result of the same experiments but performed on network architecture B. The outcomes are analogous to those achieved by network architecture A. The additional regularization in this architecture did not help the network to learn more bearing fault relevant features, possibly due to the dominance of non-defect related features in the signal which associate with a given class.

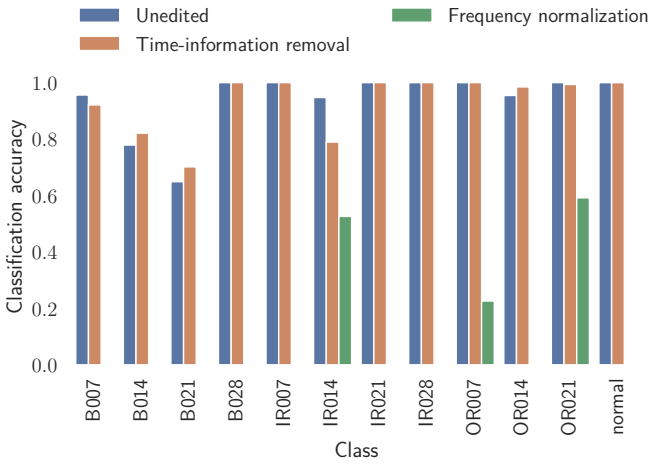


Figure 5. Classification performance (accuracy) for the signal transformations using network architecture A on the CWRU data set.

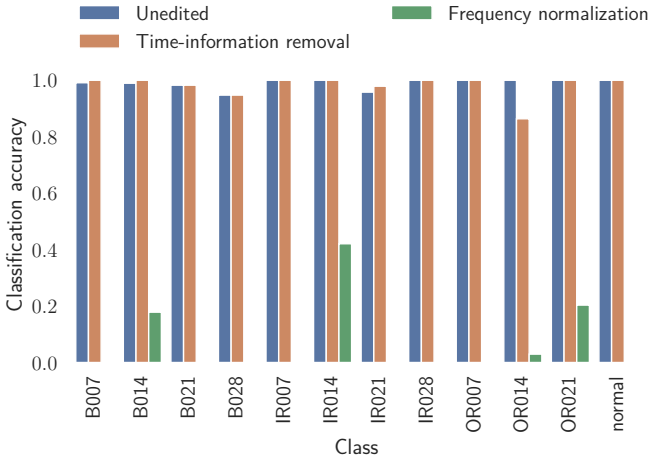


Figure 6. Classification performance (accuracy) for signal transformations using network architecture B on the CWRU data set.

3.2. Internal data set

In order to assess the applicability of our findings to other machines, the above mentioned experiments are repeated on a second data set created at SKF. This data set contains vibration signals recorded with a sampling rate of 100 kHz for a 6206 ETN9/C3 deep grove ball bearing. Rectangular defects are engraved on the outer ring of the bearings with the following sizes: 0.1 mm, 0.25 mm, 0.5 mm, 3 mm, 5 mm and 8 mm, alongside data from a bearing with no defect. The bearing is radially loaded and mounted at four distinct positions (0, 20, 40 and 60 degrees), where 0 degrees is the condition where the defect is centered in the loaded zone. The accelerometer is positioned close to the loaded zone in the vertical direction. The data is recorded under four load conditions (200 N, 300 N, 400 N and 500 N) while the rotational speed is maintained at 1500 RPM. To be consistent with the CWRU data, signals are resampled to 12 kHz.

	Network A	**Network B**
0 mm	0.000	0.000
0.1 mm	0.882	0.842
0.25 mm	0.789	0.974
0.5 mm	1.000	1.000
3 mm	0.750	1.000
5 mm	0.912	1.000
8 mm	1.000	1.000
All test samples	**0.837**	**0.880**

Table 5. Classification performance (F1-scores) of network architecture A and B for different classes.

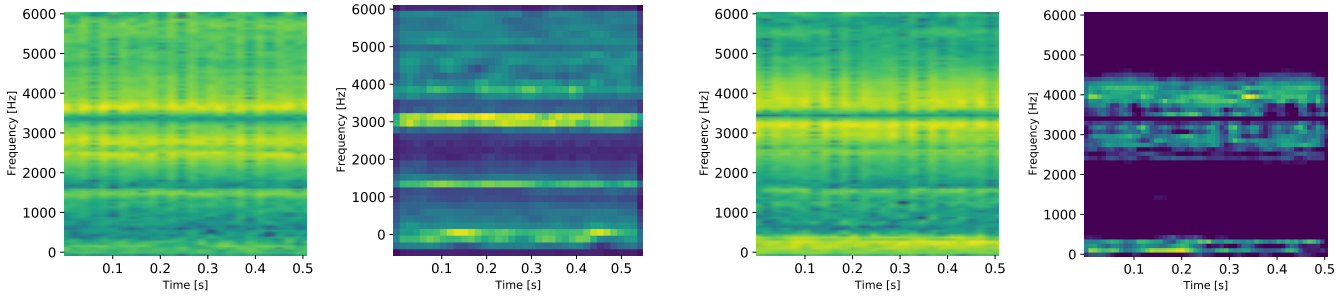
3.2.1. Performance of the networks

The classification results of network architecture A and B using this data set are presented in table 5. Both networks perform well on all classes except for the non-defect class (0 mm). Similar to the experiments with the CWRU data set, network architecture B outperforms network architecture A.

3.2.2. Interpretability

Examples of Grad-CAM maps of both network architectures are presented in figure 7. Similar to the experiments using the CWRU data set, the dependence of the network on resonance related features becomes evident. Although the two data sets are independent and acquired using two different machines, the neural networks tend to classify based on the dominant features in the data set, namely the resonances of the system.

The results of the input transformation experiments (figures 8 and 9) confirm the findings obtained on the CWRU data set. The learned features are related to the dominant resonances of the system in the signal. The performances are largely unaffected by the time-information removal experiment, while they drop drastically with frequency normalization experiment. This further supports the hypothesis that the network learns system dependent features.



(a) Input image belonging to the 0.1 mm class (network architecture A).

(b) Grad-CAM map of the 0.1 mm class using network architecture A.

(c) Input image belonging to the 0.25 mm class (network architecture B).

(d) Grad-CAM map of the 0.25 mm class using network architecture B.

Figure 7. Examples of Grad-CAM activation maps of network architecture A and B trained on the internal data set. The color scale goes from blue (low) to yellow (high).

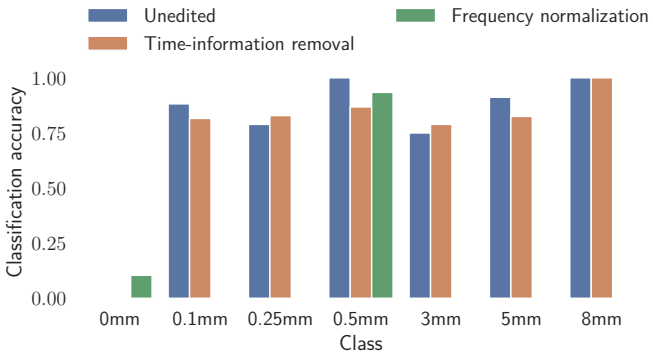


Figure 8. Classification performance (accuracy) on the internal data set for the signal transformations using network architecture A.

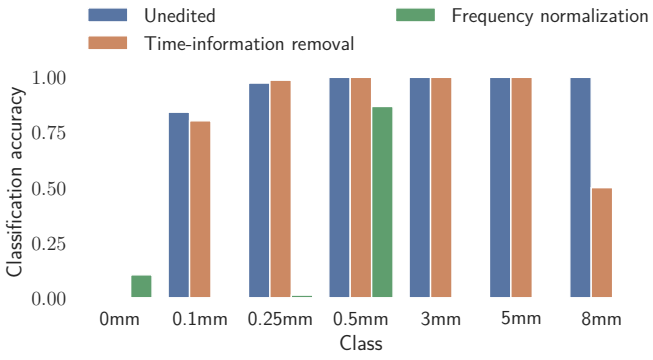


Figure 9. Classification performance (accuracy) on the internal data set for the signal transformations using network architecture B.

4. DISCUSSION AND CONCLUSIONS

Analyzing the internal representations of the network using interpretability algorithms such as Grad-CAM, together with input signal transformations, we observe that the networks are learning distinct average frequency profiles, while ignoring time-domain information. We expect that these frequency profiles are mainly dominated by the transfer function resonating frequencies. This, in turn could hint towards the issue with the generalizability of the DL approaches since the transfer function is unique between machine/sensor configuration. Note that the transfer function can still change as a function of the bearing defect (H. Zhang et al., 2021), which could explain the good performance of the network. However, this information is not generalizable to other machines.

Our findings conform with the results in Smith and Randall (2015), where it is shown that the rig assembly (a contributor to the transfer function) is affecting the vibration signal more than the bearing defect in the CWRU data set. This potentially means that the networks learn these more prominently present features in the data rather than the defect signature.

In addition to the effect of system specific features on the classification outcome, the underlying representations that caused the activation of certain classes were in particular intriguing. Informal experiments (not described in this paper) show that the network classifies random noise (2D image representation) as an outer ring defect of 21 mm length with very high confidence. In addition, the network, in some instances, classifies correctly using only the Grad-CAM representation of the signal as an input. These are very compelling observations, considering that these inputs carry no meaningful information regarding the bearing fault such as explained in, e.g., H. Zhang et al. (2021); Epps (1991).

Another interesting observation while analyzing the Grad-CAM maps was achieved by clustering these maps for a given defect according to the patterns they generated. In the CWRU data set, the patterns correlate with different sensor locations. The signals acquired from each sensor for a given defect, activate similar representations in the network. This is yet another confirmation that the networks learned transfer function related features rather than defect signature features. Moreover, it shows that the networks do not generalize simply

by providing data of the same defect acquired from different paths. Instead of learning the underlying defect features, the networks had learned multiple different transfer function representations associated to the same class.

To summarize, we conclude that the trained networks are not generalizable neither to other machines, nor to the same machine with another configuration. As a first step we propose to introduce more domain knowledge within the field of DL-based bearing fault diagnostics. This could be done for example by pre-processing the data, aiming at reducing the competing dominant machine specific features in the signal to attain a more generalizable solution. The results further highlight the importance of thorough investigation of the input signal to the neural network, the class representations learned from that input, and the generalizability of the solution.

REFERENCES

Antoni, J. (2009). Cyclostationarity by examples. *Mechanical Systems and Signal Processing*, *23*(4), 987–1036.

Chattopadhyay, A., Sarkar, A., Howlader, P., & Balasubramanian, V. N. (2017). Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. *arXiv preprint arXiv:1710.11063*.

Chen, Z., Mauricio, A., Li, W., & Gryllias, K. (2020). A deep learning method for bearing fault diagnosis based on Cyclic Spectral Coherence and Convolutional Neural Networks. *Mechanical Systems and Signal Processing*, *140*, 106683.

Epps, I. K. (1991). *An investigation into vibrations excited by discrete faults in rolling element bearings* (Unpublished doctoral dissertation). University of Canterbury. Mechanical Engineering.

Ganin, Y., & Lempitsky, V. (2014). Unsupervised domain adaptation by backpropagation. *arXiv preprint arXiv:1409.7495*.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436–444.

Liu, C., Mauricio, A., Qi, J., Peng, D., & Gryllias, K. (2020). Domain adaptation digital twin for rolling element bearing prognostics. In *Annual conference of the phm society* (Vol. 12, pp. 1–10).

Liu, H., Li, L., & Ma, J. (2016). Rolling bearing fault diagnosis based on stft-deep learning and sound signals. *Shock and Vibration*.

Randall, R. B., & Antoni, J. (2011). Rolling element bearing diagnostics—a tutorial. *Mechanical Systems and Signal Processing*, *25*(2), 485 - 520.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2019). Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, *128*(2), 336–359.

Smith, W. A., & Randall, R. B. (2015). Rolling element bearing diagnostics using the case western reserve university data: A benchmark study. *Mechanical Systems and Signal Processing*, *64*, 100–131.

Tao, H., Wang, P., Chen, Y., Stojanovic, V., & Yang, H. (2020). An unsupervised fault diagnosis method for rolling bearing using stft and generative neural networks. *Journal of the Franklin Institute*, *357*(11), 7286–7307.

Verstraete, D., Ferrada, A., Droguett, E. L., Meruane, V., & Modarres, M. (2017). Deep learning enabled fault diagnosis using time-frequency image analysis of rolling element bearings. *Shock and Vibration*.

Wang, Q., Michau, G., & Fink, O. (2019). Domain adaptive transfer learning for fault diagnosis. In (pp. 279–285).

Zhang, H., Borghesani, P., Smith, W. A., Randall, R. B., Shahriar, M. R., & Peng, Z. (2021). Tracking the natural evolution of bearing spall size using cyclic natural frequency perturbations in vibration signals. *Mechanical Systems and Signal Processing*, *151*, 107376.

Zhang, S., Zhang, S., Wang, B., & Habetler, T. G. (2020). Deep learning algorithms for bearing fault diagnostics—a comprehensive review. *IEEE Access*, *8*, 29857-29881.