# Transfer Learning for Anomaly Detection Using Bearings' Vibration Signals

**Diego Nieves Avendano, Dirk Deschrijver and Sofie Van Hoecke**

*IDLab, Ghent University - imec, Technologiepark-Zwijnaarde 122, 9052 Gent, Belgium.*
E-mail: diego.nievesavendano@ugent.be

(Received 22 June 2023; accepted 9 October 2023)

Predictive maintenance is becoming increasingly important in the industry. Despite considerable advances in data collection and data-driven models, there are still limitations when deploying models in practice. One of the main limitations is the large datasets required to train these models. As a potential solution, transfer learning can be used to reuse knowledge acquired from large datasets for similar tasks under different conditions. This paper investigates the transferability of the specific anomaly detection scenario, an unsupervised learning task with heavily imbalanced label distribution. This paper uses a dataset generated from a bearing test platform in which bearings are run until failure under different operating conditions. A lightweight deep learning model, MobileNetV2, is employed to create a baseline model capable of detecting anomalies for a specific working condition. The model is then adapted using transfer learning to identify anomalies under new operating conditions with limited data accurately. The results show that the data for new conditions is insufficient to train an adequate model and that transfer learning can overcome this limitation. The adapted models can detect anomalies before the expert's knowledge reference value. Although this shows that transfer learning can detect anomalies earlier, the results must be evaluated carefully to avoid false positives. While anomaly detection aims to identify changes in feature distributions, transfer learning aims to align different feature distributions. Transfer learning for unsupervised learning has rarely been explored. To the best of our knowledge, this is one of the few works addressing it in the context of predictive maintenance for anomaly detection.

## NOMENCLATURE

| | |
|---|---|
| **AD** | Anomaly Detection |
| **AE** | Auto-Encoder |
| **ANN** | Artificial Neural Network |
| **CNN** | Convolutional Neural Network |
| **CWRU** | Case Western Reserve University. Dataset abbreviation, used as they areusually more known by their abbreviation rather than the full name. |
| **IMS** | Intelligent Maintenance System. Dataset abbreviation, *idem*. |
| **LSTM** | Long Short-Term Memory Network |
| **MNv2** | MobileNetV2 |
| **PACS** | Photo-Art-Cartoon-Sketch. Dataset abbreviation, *idem*. |
| **PdM** | Predictive Maintenance |
| **PHM** | Prognositcs and Health Management |
| **RNN** | Recurrent Neural Networks |
| **RPM** | Revolutions Per Minute |
| **RUL** | Remaining Useful Life |
| **SGD** | Stochastic Gradient Descent |
| **SMLL** | Smart Maintenance Living Lab. Introduced dataset by our research. |
| **STFT** | Short-Time Fourier Transform |
| **TL** | Transfer Learning |

## 1. INTRODUCTION

Predictive maintenance (PdM) models of mechanical components are crucial for Prognostics and Health Management (PHM) within the industry. Accurate PdM models contribute to quality, safety, maintenance scheduling, and cost reduction improvements. Anomaly detection (AD) most commonly refers to the unsupervised task of detecting events that deviate from normal operating conditions. This paper examines the use of artificial neural networks and transfer learning on vibration data from bearings under different operating conditions (speed changes) for anomaly detection. The methodology is evaluated using the Smart Maintenance Living Lab (SMLL) dataset by Flanders Make and imec.[1]

While deep learning techniques have been successfully used for fault identification and classification, the application of transfer learning (TL) in anomaly detection has been limited. This fact is unsurprising given that AD is an inherently challenging task due to the frequent lack of ground truth and being an imbalanced, unsupervised task. In addition, AD is, on its own, responsible for detecting shifts in the data features that correspond to unexpected behavior, while TL is responsible for consolidating data with different feature distributions as normal behaviors. In other words, a straightforward AD approach would label samples coming from new domains as anomalous. Therefore, the challenge is finding a representation that maps abundant information from the source to the sparse target domain for improved anomaly identification in both domains.

This paper is divided as follows: Section 2 reviews existing literature on PdM tasks using deep learning and transfer learning techniques, focusing on fault classification and anomaly detection. Section 3 introduces the SMLL dataset. Section 4 outlines the methodology, which consists of the data processing, the details on the baseline model, and how the model is adapted to the different operational conditions. Section 5 presents the results and discussion of the results obtained in the baseline and the transfer learning generalization performance. Section 6 proposes future research work based on the insights gained. Finally, section 7 presents the conclusion.

## 2. RELATED WORK

Physics-based and statistical models have previously been used for the analysis of vibration signals of bearings for tasks such as detecting degradation onset, fault detection and classification, and estimation remaining useful life (RUL).[2] However, in recent years, deep learning techniques have gained attention due to their improved performance, ability to discover features from raw or lightly processed sensor data, capability to model complex processes without prior domain knowledge, and require minimal assumptions about feature distributions.[2,3] Moreover, deep learning models offer advantages in transfer learning owing to their inherent ability to learn complex representations that can be generalized across different contexts. In addition, they possess high transferability across considerably different tasks and have been proven to boost generalization on new tasks when starting from a pre-trained model.[4]

This section presents an overview of the deep learning literature concerning general PdM tasks. Subsequently, it gives a literature review of transfer learning in the context of PdM. Finally, it discusses the advantages of using lightweight deep learning models.

### 2.1. Deep Learning in PdM Tasks

To solve PdM tasks, deep learning techniques, specifically Artificial Neural Networks (ANNs), have been employed to analyze raw time signals to predict different fault types of some of the most popular bearing datasets, such as the Case Western Reserve University (CWRU) and the Intelligent Maintenance System (IMS) Bearing Dataset.[5] However, using dense layers over temporal data is only possible with relatively small time windows, as the dimensionality of the network increases drastically with high sampling rates. Different approaches that reduce the signal complexity can be used to solve this limitation, such as representation learning or transformations of the vibration signal.

Like dictionary learning, representation learning techniques allow learning complex signals in reduced spaces via linear combinations of prototype signals called atoms. These sparse dictionary learning techniques have been combined with data augmentation techniques for diagnosing planetary bearings.[6] Alternatively, transforming vibration data from the time domain into the frequency domain allows for compact inputs and can prevent the neural networks from being overparametrized. Moreover, spectral representations remove the need to correctly detect and align the carrier signal's peaks and the characteristic frequency components, which would otherwise require the alignment to extract the correct features in the input layer of the neural network. Spectrogram representations have been used with feed-forward ANNs,[7] self-organizing maps,[8] and other deep learning architectures.[9]

Convolutional Neural Networks (CNN) are an evident model choice given the time and frequency relations between adjacent samples. CNNs are characterized by their ability to discover structural information in the data and are more parameter-efficient than ANNs. CNN architectures have been used over temporal data in many forms, such as the raw vibration signals[10–12] and filtered and pre-processed signals.[13] In some cases, time series can be treated as a particular case of two-dimensional convolutions, either by splitting a time window into segments and using each segment as a separate channel[14] or by rearranging the segments and processing them as two-dimensional images.[5,15] However, it is important to notice that these approaches can be computationally expensive as machines operate at high frequencies that must be represented in large input vectors to be meaningful. Furthermore, the temporal representation is sensible to the location of the peaks of the carrier signal. This limitation can be alleviated by using pooling operators to induce translation invariance. However, this is only possible if there is a sufficient amount of pooling layers, which may come at the expense of increased model complexity.

In contrast, using the spectral representation, there are no limitations concerning the carrier's location and characteristic frequency components. Another two-dimensional representation is the frequency spectrogram over time.[16–18] The representation used in their research aligns with the one selected in this study. Other frequency-time representations, such as the wavelet decomposition, have also proven successful.[19–22] Finally, architectures focused on temporal relations, such as Recurrent Neural Networks (RNNs) and Long Short-term Memory (LSTM), are also used due to their inherent capacity to learn from data with long-range temporal dependencies.[23–27] These techniques can again be applied over temporal data or temporal-frequency decomposition. The literature review generally points to a preference towards frequency representations as they reduce the model's complexity and can preserve the vital information for correct diagnosis.

In specific PdM tasks, CNNs have been investigated for the creation of health indexes, which are unit-less metrics that assist in condition monitoring,[28,29] estimating remaining useful life,[14,30,31] fault classification,[30,32] as well as anomaly detection.[24,31] Furthermore, RNNs and LSTMs have been investigated for remaining useful life estimation,[23,26,33] fault classification,[25] as well as anomaly detection.[24,27] The field has considerable interest, which has increased in recent years. More extensive reviews can be found in recent field surveys.[9]

### 2.2. Transfer Learning in PdM Tasks

Transfer learning addresses problems in which the objective is reusing previously acquired knowledge for similar tasks. The TL problem can be phrased as follows: (i) given a group of Domains $\mathcal{D}$, each domain contains a dataset with a given set of features; (ii) the features in each domain measure the same properties or similar ones, but the distributions across domains differ; and (iii) TL finds a way to leverage the common information across domains to improve results on the task for any given domain. Most commonly, TL is viewed as a problem in which, for a first domain, there is a considerable amount of (high quality) data, and the knowledge of this domain wants to be applied to a new domain, where there is a limited amount of data, and sometimes of lower quality. In other cases, it can be seen as reusing a model trained to solve a specific task (e.g., image recognition) to solve a secondary task (e.g., semantic segmentation).

Two common approaches in transfer learning for pre-trained models are stacking and fine-tuning. In both methods, a neural network is trained with a large amount of high-quality data from the source domain. Afterward, this trained model is adapted for a secondary task by stacking or fine-tuning. Stacking is an approach in which a mid-layer output of the model is used as the input of a new machine learning model. In fine-tuning, the baseline model is further trained using the data from the new domain while restricting weight adaptation to the last few layers. Additionally, a third scenario exists, where a small number of layers are appended after embedding extraction. Combined with a subset of the baseline model's layers, these are fine-tuned.

Large deep learning models, such as ResNet[34] and InceptionV3,[35] were initially developed for image classification tasks and trained on the Imagenet dataset,[36] which consists of real-world object images. However, the trained versions of these models have been reused for different tasks with different data representations. For example, fault classification of wind turbine gearboxes using wavelet representation as inputs and the pre-trained ResNet as feature map extraction;[32] fault detection in photovoltaic plants using thermal images and the pre-trained ResNet;[37] and wear estimation of cutting tools based on image inputs using pre-trained models such as ResNet, InceptionV3 and AlexNet as basis for a fine-tuning approach.[30] It is important to consider the validity of pre-trained weights in these approaches, as they have been trained to recognize objects in real-life images. However, this may not directly translate to representations of spectrograms or other types of images, such as the physical condition of mechanical components.

Regardless of whether using real-life image-trained networks is a valid approach, a more common problem is the transfer learning task within two predictive maintenance tasks. This shift can occur in cases where the source and target correspond to similar components with different specifications;[10] the analysis of different fault types;[38] or discrepancies in data distributions of the same mechanical component due to significant heterogeneity.[27,39] Overall, the coverage of studies on transferal with the end goal of anomaly detection has been limited.

To the best of our knowledge, there are limited studies of anomaly detection research in the context of transfer learning (AD-TL). One such work is presented by Michau et al.,[40] where they detect anomalies in simulated engines operating under different conditions from the Turbofan jet engine dataset and faulty bearings from the CWRU dataset. Their approach consists of input alignment using adversarial deep learning. The main advantage of their method is the possibility of doing the transferal in an unsupervised way, removing the need to collect labels in the target. Their results point to an accuracy of up to 100%. However, the authors specify that this is only achievable if the number of source and target data points is the same. There is a considerable imbalance between source and target domains in many transfer scenarios, including those presented in this paper.

Another example of AD-TL uses Siamese networks in image recognition tasks.[41] Their research is conducted on hyperspectral images, and the end goal is to detect pixel anomalies over satellite pictures. To achieve this, the deep learning model is trained in a self-supervised approach in which random pixels are evaluated as originating from the same image. The pixels of a figure are compared against their neighbors to detect anomalies, and a likelihood score is obtained through the Siamese network. This approach does not apply to the present research as spectrograms are not translation invariant like real-world images.

Compared to previous work, the task of AD-TL in the context of PdM presents the following differences: Due to the costs associated with generating samples, (i) source datasets tend to be considerably smaller than those seen in other domains such as image classification; and (ii) the ratio of samples between source and target domain is heavily imbalanced. Traditional TL benchmarks have more balanced ratios between source and target domains. For example, in the office dataset, the ratios range between 1.87 to 6.59;[42] for the extended office-Home dataset, the ratios are close to 1[43], and for the Photo-Art-Cartoon-Sketch (PACS) dataset the ratios range from 1.14 to 2.35.[44] In the presented Smart Maintenance Living Lab (SMLL) dataset, the ratio ranges from 8.6 to 21.6, meaning less data is available for each target domain. Finally, (iii) there is considerable heterogeneity in the source domain, which makes non-TL tasks already hard to generalize, as it has been extensively reported in similar datasets such as Pronostia,[45] IMS[46] as well as SMLL.[47] Although this last issue can also be the case in image classification tasks, there is no clear way of comparing degrees of heterogeneity for this purpose.

So far, the works of TL applied to the domain of PdM have been limited and, when available, have focused on the supervised fault classification task. Other areas, such as RUL (supervised regression) and AD (unsupervised), which are inherently more challenging, have received limited attention in TL research. The authors consider that this may be due to the limited number of large datasets for training reliable baseline models and the challenge posed in detecting anomalies or predicting RUL in inherently dissimilar datasets.

## 2.3. Lightweight Models and MobileNet

Since the middle of the 2010s, and with the advent of successful deep learning models, the case has been made in favor of lightweight deep learning. Although no formal definition exists for a lightweight model, it is often presented as an optimization for well-known large deep-learning models. The growing interest in lighter models comes from the fact that a lower number of parameters reduces training and inference time. Furthermore, it has also been demonstrated that smaller models can increase performance and generalization. For example, when LeNet-5 was first introduced in 1995, it consisted of 1 million parameters and was the cornerstone in character recognition.[48] Progress over the following years would be mainly achieved by increasing the network size. In 2014, VGG-16 was introduced[49]; it consisted of 138 million parameters and was, at its time, the best-performing model for the ImageNet dataset. Others quickly outperformed the model, such as ResNet in 2015 with 60 million[34] and SimpleNet in 2016 with only 6.4 million parameters.[50] This highlights the trend of reducing the number of parameters by improving the architecture rather than increasing the number of parameters.

The concept of lightweight has been addressed in regards to different aspects, such as reducing the number of parameters of well-known models by pruning the weights[51,52], compressing the model weights by using hashing functions[43], reducing the numerical precision of the learned weights[53], among other techniques.[54] However, most of these approaches focus on reducing inference time after training a large network. In contrast, different approaches have focused on creating architectures that are also easy to train, such as SqueezeNet,[55] Simplenet,[50] NASNet[56] and the Mobile Networks (MobileNet).[57] This work focuses on using MobileNets, although other architectures could be considered for the same methodology presented here. MobileNets are a family of architectures that enable computer vision applications for mobile devices. To achieve this, the architectures use a considerably lower number of parameters and replace the standard convolutional operator with a depth-wise separable convolution. Separable convolutions are notorious for enabling a reduced number of multiplications while still performing in pairs with state-of-the-art models for object detection.

The use of lightweight models in PdM is relevant to today's industrial applications as it allows monitoring equipment on edge devices without the need for GPUs. Recent research has proven its advantage over large deep-learning models in PdM tasks. For example, a MobileNet V2 architecture was extended to include a channel attention mechanism and proved successful in the fault-bearing diagnosis of the CWRU dataset.[58] Their results proved to be more accurate (80.60%) and faster than deep architectures such as AlexNet and Res-Net-50 with accuracies of 58.6% and 78.78%, respectively. In addition, the model reduction is smaller in memory and has faster inference times. Other publications have achieved even higher performance on the CWRU dataset using an earlier version of the network, namely the MobileNet V1, with an accuracy of 96%.[59] Some approaches have also considered lightweight models in combination with weight pruning. For example, the MobileNet V2 with weight pruning was evaluated for bearing fault diagnosis under different degrees of sparsity. Their study found a minimal decrease in the model's accuracy despite considerable pruning; in fact, a moderate amount of pruning proved to give the best results.[60] Their results compare the diagnosis using LeNet-5, 98.74%, against the original MobileNet V2, 99.58%, and MobileNet V2 with a sparsity of 0.3, 98.95%. The dataset used is collected by themselves. Therefore, the results cannot be compared to other benchmarks. These results show that light models are well suited for PdM tasks, provide considerable benefits for real applications that require low computation, and, in some cases, provide better results than standard models.

**Table 1.** Model comparison across popular deep learning architectures. Parameters and Multiply-Add Accumulate (MACC) are reported based on the respective benchmarks while inferring for the ImageNet dataset. For the model presented here no estimation of the MACC is provided as it is not intended to be used with ImageNet and input size is a crucial parameter for determining the number of parameters and required computations.

| Network | Year | Parameters | MACC |
|---|---|---|---|
| AlexNet[50] | 2012 | 60.97 M | 7.27 G |
| VGG16[50] | 2014 | 138.36 M | 154.7 G |
| ResNet-152[50] | 2015 | 60.19 M | 11.3 G |
| SimpleNet[50] | 2016 | 6.4 M | 1.9 G |
| SqueezeNet[50] | 2016 | 1.25 M | 861 M |
| NASNet-A (4@1056)[56] | 2017 | 5.3 M | 564 M |
| NASNet-A (3@960)[56] | 2017 | 4.9 M | 558 M |
| MobileNet V2 1.0[64] | 2018 | 3.4 M | 300 M |
| MobileNet V2 0.35[64] | 2018 | 1.6 M | 59.2 M |
| MobileNet V3 Large 1.0[64] | 2019 | 5.4 M | 219 M |
| MobileNet V3 Large 0.75[64] | 2019 | 4.0 M | 155 M |
| MobileNet V3 Small 1.0[64] | 2019 | 2.5 M | 56 M |
| MobileNet V3 Small 0.75[64] | 2019 | 2 M | 44 M |
| Mini MobileNet V2 (ours) | - | 26 K | NA |

To the best of our knowledge, transfer learning using MobileNets has only been used in the context of image recognition tasks. For example, for evaluation of asphalt quality,[61] or steel frames' damage.[62] The pre-trained MobileNet uses the real ImageNet dataset, and as discussed in Section 2.2, these weights should not be expected to provide any information for downstream tasks that use representations such as spectrogram. Due to this, the authors considered that there is limited work on using MobileNet for spectrogram analysis.

Finally, Table 1 compares the number of parameters and computational complexity for popular image recognition models. Figure 1 visually compares different models' performance in the ImageNet classification task. Notice that the selected architecture for this paper, namely the MobileNet v2, is a high-speed and compact model, and it forms part of the Pareto front for optimal model selection. For a broader array of more recent image recognition models, a comprehensive survey can be found in the work by Bianco et al.,[63] although it mainly includes models with even higher complexity. Importantly, it must be emphasized that these comparisons serve as reference points and may not directly indicate the most suitable model choice for either anomaly detection or transfer learning.

## 3. SMART MAINTENANCE LAB DATASET

The Smart Maintenance Living Lab (SMLL) is an open test research platform that aims to support the adoption of condition monitoring technologies.[1] The platform consists of a fleet of seven identical drive-train setups that perform accelerated lifetime tests on FAG 6205 bearings. The fleet offers two advantages: first, it allows faster data collection; and second, since identical drive-train systems can have variability, it provides the opportunity for training and evaluating robust models.[1,66] The data is rich in the variety of operating conditions,

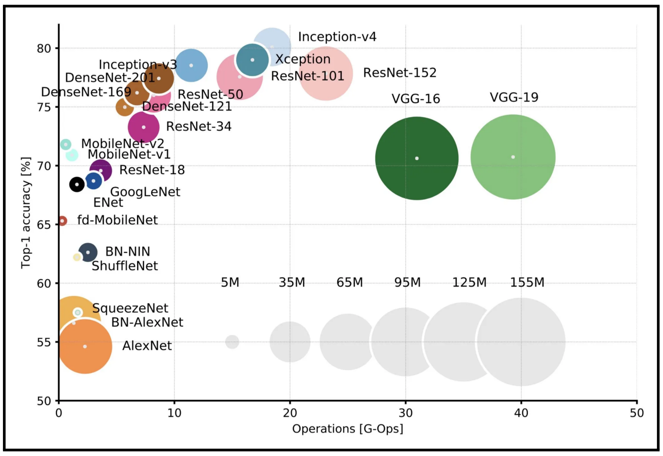

**Figure 1.** Top-1 accuracy vs. operations (inference) and network size for selected architectures. The size of the blobs is proportional to the number of network parameters. Figure by Canziani et al.[65]

such as speed and loads, in addition to speed changes while operating. This fact is a considerable improvement against most available benchmarks, which tend to be limited to a dozen or fewer tests and, in most cases, have a limited number of operating conditions.

As of the time of writing, the SMLL dataset [1] consists of 145 bearing tests. However, data collection is ongoing.

## 3.1. Data Collection

Tests are performed under different speeds, loads, and initial conditions. These variations are meant to represent different possible operating conditions. The speed of a test can either be constant or follow a saw-tooth profile. Constant speeds are set to a fixed value between 1500 and 2100 rpm. The speed varies from 1000 rpm to 2000 rpm in the saw-tooth profiles. The test starts at a speed of 1000 rpm and is increased in steps of 100 rpm. Each speed is kept constant for 60 s, and once the speed reaches 2000 rpm, the speed is set back to 1000 rpm. The saw-tooth profile tests were captured to obtain more variability in the data for each test. The load of the tests is fixed to a value between 8 and 9 kN. Finally, the initial condition indicates whether the bearing is indented at the start of the test. The indentations of the bearings are meant to accelerate degradation. The indentations are done in the inner race using a Rockwell C hardness tester and have diameters within $400 \pm 25 \mu m$. These indents are small enough that the bearing can be considered healthy at the beginning of the test but significant enough to guarantee that the degradation onset occurs within a few hours. Notice that the assumption here is that indented bearings behave as healthy bearings at the beginning of the test. However, it is important to mention that already clear differences exist in the spectrogram content. Despite this, there is a considerable long steady state in which the vibration profile does not change, indicating no further degradation on the bearing. The outcomes of each test are also reported, which can be either that the bearing remains healthy or damage is detected. Each report contains information concerning the condition of the bearing components, in addition to a report of events during the tests and photos of the end condition of the rolling elements, rings, and axles of the setup.

**Table 2.** Sample size in hours of recording for each speed.

| Speed | Train | Evaluation | Domain |
|---|---|---|---|
| 1000 | 2.3 h | 4.2 h | Source |
| 1100 | 2.5 h | 4.6 h | Target |
| 1200 | 2.6 h | 4.6 h | Target |
| 1300 | 2.5 h | 4.7 h | Target |
| 1400 | 2.5 h | 4.7 h | Target |
| 1500 | 2.5 h | 4.6 h | Target |
| 1600 | 2.5 h | 4.6 h | Target |
| 1700 | 2.4 h | 4.6 h | Target |
| 1800 | 2.4 h | 4.6 h | Target |
| 1900 | 2.4 h | 4.6 h | Target |
| 2000 | 48.3 h | 154.2 h | Source |

Tests are conducted using vibration sensors. The vibration signal is sampled at 50 kHz. In early tests, a sample was collected every 10 s, while the samples were collected every second for later trials. Most of the tests correspond to the second case.

The stop condition differs depending on whether the bearing is indented or not. In the case of unindented bearings, this is defined as the moment when the temperature stabilizes and at least two hours have passed. Unindented bearings are generally not expected to fail as their expected lifetime exceeds the testing period. During the evaluation of these bearings, no anomalies were reported, and no damage was found during the examination of their conditions. For the indented bearings, the stop condition is when the peak vibrations reach a magnitude of 20 g. This condition is not reached for a small number of bearings, and tests are stopped after a few hours. These bearings undergo degradation, and anomalies are reported, but the tests had to be prematurely stopped for safety reasons.

Figure 2 shows examples of damaged bearings after the accelerated life tests.

## 3.2. Source and Target Datasets

This research focuses on anomaly detection under the transfer between different operating conditions. The scenario involves training an anomaly detection algorithm to perform well at a specific speed and exploring its performance under a different speed with limited sample availability. In this research, the evaluation focuses solely on tests conducted with a load of 9 kN, as it is the load with more speed-varying data available. The source domain corresponds to the configurations with a load of 9 kN and speeds of 1000 and 2000 rpm, which correspond to the minimum and maximum operational speeds. The source dataset contains both indented and non-indented bearings. The target dataset corresponds to tests with a load of 9 kN and speeds between 1000 and 2000 rpm so that the transfer learning step corresponds to a speed interpolation. Table 2 summarizes each speed's recorded hours.

The source dataset consists of 72 bearing recordings, of which 64 are tests performed with a load of 9 kN and 2000 rpm, and eight are tests performed with the saw-tooth speed pattern. From this data, 13 bearings are used as validation data for parameter tuning and prevention of overfitting. From the saw-tooth speed profile recordings, only the segments at the lowest

[1] Dataset is available upon request at Flanders Make website https://www.flandersmake.be/en/datasets

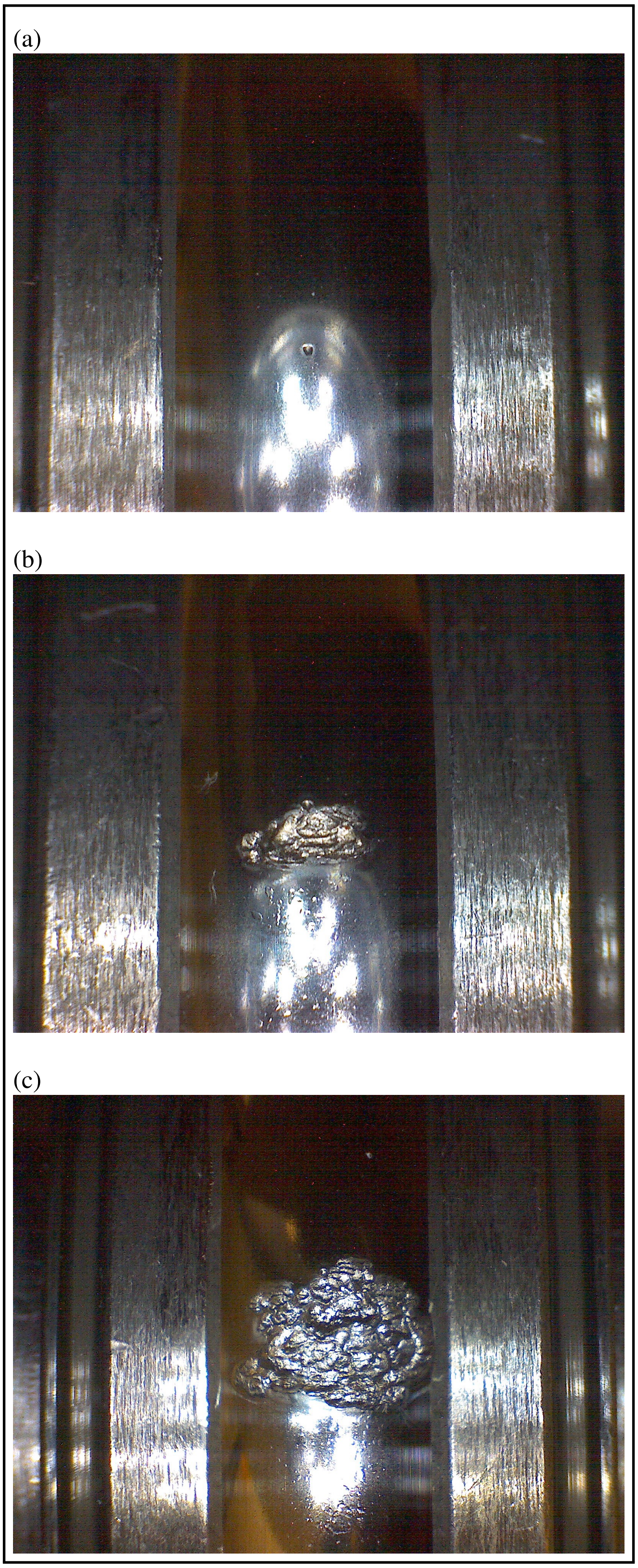

**Figure 2.** Details of the bearings' condition before and after testing. a) Bearing A7 with a small indentation to induce accelerated degradation. The indentation is located at the inner race and has a diameter of approximately 400 ± 25 $\mu m$; b) Bearing A43 with a moderate amount of damage after the test; c) Bearing A47 with extensive damage after the test. For reference, the inner race diameter of the FAG 6205 is 25 mm.

**Table 3.** Summary of the dataset and experiment conditions.

| | Dataset | |
|---|---|---|
| | Source | Target |
| Bearing type | FAG 6205-C-TVH | |
| Initial condition | - Healthy<br>- Indented (400 ± 25 $\mu$m) | |
| Stop condition | - Healthy: 2 hrs after stable temperature<br>- Indented: Vibrations exceeding 20 g or after several hours without failure. | |
| Sampling rate | 50 kHz | |
| Acquisition frequency | - Every second<br>- Every 10 seconds | |
| Speed pattern | - Fixed<br>- Sawtooth | - Sawtooth |
| Operating speeds (rpm) | 1000, 2000 | 1100, 1200, 1300<br>1400, 1500, 1600<br>1700, 1800, 1900 |
| Operating load | 2 kN | |
| Number of tests | - Fixed speed: 64<br>- Sawtooth profile: 8 | - Fixed speed: 0<br>- Sawtooth profile: 14 |

speed (1000 rpm) are used as part of the source dataset.

The target dataset consists of 14 bearing recordings, all with a saw-tooth speed profile. From these, eight correspond to the same bearings in the source dataset but for the segments where the speed differs from 1000 or 2000 rpm. In addition, six completely separated tests are included. Table 3 summarizes the source and target datasets.

## 4. METHODOLOGY

### 4.1. Spectrograms and Mel Spectrograms

The vibration signals are transformed into spectrograms using the short-time Fourier transform (STFT) and log transformation to obtain the mel spectrogram. A spectrogram represents the frequency content over time for a given signal. The mel spectrogram converts a logarithmic scale to the high frequencies while preserving a linear scale for the lower frequencies. The mel spectrogram representation reduces the frequency components to a defined number of mel-frequencies. After initial tests, the mel spectrogram representation was preferred due to its compressed frequency dimension, as most faults are found within the low-frequency range. In general, reducing the number of input features allows deep learning models to require fewer parameters. Figure 3 shows an example of the signal in time and the two frequency representations. Finally, the log mel energy of the spectrogram is calculated and used as the features. Notice that in the mel spectrogram representation, the higher contrast between adjacent frequencies is due to the logarithmic scaling.

### 4.2. Anomaly Detection

This research aims to train models that can detect anomalies, specifically the first anomaly during a test, which often corresponds to the degradation onset. This fact can be beneficial for predictive maintenance tasks, such as better Remaining Useful Life (RUL) estimations and optimizing maintenance schedules. The AD task must be unsupervised, meaning no labeling should be assumed. To accomplish the task, the models are trained to predict the operating speed over healthy data, or in

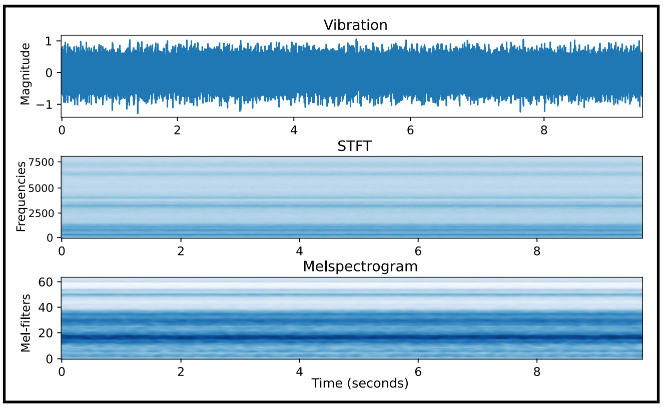

**Figure 3.** Vibration signal in time and its representations as spectrogram and mel spectrogram.

the case of indented bearings, the beginning of the test, which can be considered healthy. When bearings start to degrade, the models will predict erroneous speeds as this is unseen data. The difference between the predicted speed value and the actual speed can be seen as a health indicator that can be used for AD. This approach is similar to the more commonly used auto-encoder (AE) method in which the original signal is compressed and later reconstructed. In the AE case, unseen inputs propagate as data points distant from the embedding centroids, which in turn causes significant reconstruction errors. Analogously, the difference between the predicted and actual speeds is used instead. Although data labels are used to train the network, anomaly detection remains unsupervised as no information about these events is provided during training.

The target of the neural network corresponds to the min-max normalized speed value. In the baseline model, this corresponds to either 0.5 for 1000 rpm or 1 for 2000 rpm. For the target domain, the speeds are between 0.5 and 1.

To detect anomalies, an error threshold needs to be defined. In supervised AD, this is achieved by optimizing a threshold value using the labeled example. For example, thresholds can be defined based on a percentile of the reconstruction error over several complete tests, e.g., 95% of the upper value, or based on visual examination of the temporal trends of the reconstruction error. Other options for the supervised scenario can use asymmetric cost functions that enforce greater importance on correctly classifying anomalies. Due to the nature of this research, supervised approaches to determine the threshold cannot be used; instead, the threshold is defined in an online approach. The threshold is defined for each test based on the reconstruction error over the training period or the first 25 minutes for hold-out test bearings. The threshold is defined as the sum of the 95th percentile ($P_{95}$) plus the 10th percentile ($P_{10}$). The $P_{95}$ value was selected to avoid outliers that may occur during the startup period of the test.

## 4.3. MobileNet V2

All models follow the MobileNetV2 (MNv2) architecture.[67] This architecture uses depth-wise separable convolutions and connected residuals over the bottlenecks, allowing inducing meaningful representations. The key component of MNv2 is the bottleneck unit presented in Fig. 4. Each bottleneck unit consists of three sub-units: an expansion layer that extracts features from the input, expanding the input to a given number of filters defined by the expansion parameter; a depth-wise convolution layer that convolves each of the expanded outputs separately; and a projection layer that reduces redundant information. This architecture is selected due to its fast training and successful use in image classification tasks.

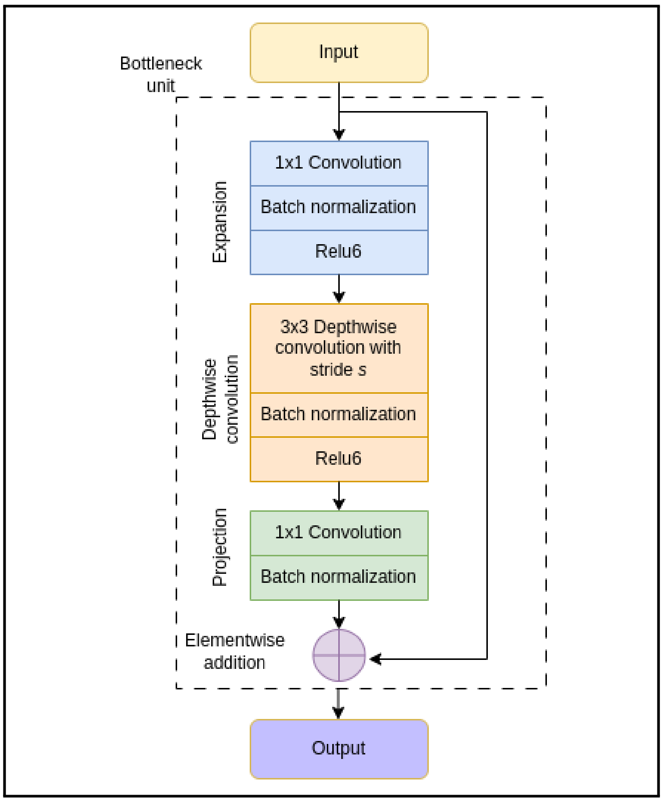

**Figure 4.** Bottleneck units of the MNv2 are composed of three stages, namely an expansion, a separable convolution and a compression stage (projection and element-wise addition).

The original MNv2 model is trained to recognize real-life images.[67] However, the problem at hand uses spectrograms as input images to the network, and instead, the weights were initialized randomly. In some cases, MNv2 or similar architectures are reused using the weights learned from training on imagenet.[16] This is an essential difference as we consider that the patterns generated by the convolutional filters are likely to differ between real-life images and spectrograms. Finally, the original MobileNetV2 consists of approximately 3,400,000 parameters, with differences depending on the selected width parameter. The many parameters allow MobileNetV2 to learn a broad range of images; however, the dataset at hand is considerably smaller than image recognition datasets. To improve training time and generalization performance, a smaller version of the MobileNetV2 architecture is used; this consists of a lower number of layers and less convolutional kernels. The proposed architecture consists of approximately 26,000 parameters. Table 4 describes the architecture, where each bottleneck corresponds to a 3-layer unit as illustrated in Fig. 4. Notice that the column *n* corresponds to the number of units, which in this case is kept constant to 1, whereas the original MobileNetV2 uses multiple blocks consecutively.

**Table 4.** Reduced MNv2 architecture. Here *c* corresponds to the number of channels, *n* the number of times the given layer is repeated, and *s* the stride.

| Operator | *c* | *n* | *s* |
|---|---|---|---|
| Conv2D (3×3) | 16 | 1 | 2 |
| Bottleneck | 32 | 1 | 1 |
| Bottleneck | 64 | 1 | 1 |
| Bottleneck | 64 | 1 | 1 |
| Conv2D (3×3) | 128 | 1 | 1 |
| Global Avg. Pooling | - | 1 | - |
| Dense | - | 1 | - |

**Table 5.** Grid search parameters.

| | Parameter | Range |
|---|---|---|
| Optimizer | Batch size | 64, 128, 256 |
| parameter | Learning rate | 0.01, 0.005, 0.001 |

## 4.4. Baseline Model

The baseline model is an MNv2 model trained to achieve high efficiency in the source domain, specifically the tests performed at 1000 and 2000 rpm. For each test, only the first 35% of the samples are used for training or validation, with a maximum of 2000 samples. This segmentation is based on the assumption that indented bearings can be considered healthy during this period. The same assumption also applies to unindented bearings, although considering the slow degradation process, it would be reasonable to assume an even longer period. However, this is not considered in the current work. Based on this segmentation, the remaining samples in a run are used as test data, with a limited degree of information leakage due to serial correlation.

The optimizer used is Stochastic Gradient Descent (SGD) with learning decay and early stop to prevent overfitting. The optimizer's parameters and data input representation are selected using grid search over the ranges shown in Tables 5 and 6, respectively. The selection is based on repeated 5-fold hold-out validation. The best parameters are a batch size of 128, a learning rate of 0.001, a temporal width of 49, and 64 mel frequencies. It is worth noting that the differences in data representation (Table 6) and the batch size had a negligible effect.

## 4.5. Transfer Learning by Fine-tuning

The fine-tuned model uses the embedding layer after the global average pooling of the baseline model (see Table 4), followed by a dense layer of 64 units with ReLU activation. The baseline model is fine-tuned using the target domain data, obtaining an adapted model for each new target speed. The models are trained using backpropagation with the same parameters as the baseline model. For this purpose, 11 bearings are used for training and 3 for validation, followed by a split of 78% for training and 22% for validation. Similar to the training of the baseline model, only the first 35% of run samples (up to a maximum of 2000 samples) are used for training, while evaluation is carried over the rest of the run. The fine-tuning uses the best hyperparameters found while tuning the baseline model, employing SGD as an optimizer with learning decay and early stop.

**Table 6.** Data representation parameters.

| | Parameter | Range |
|---|---|---|
| Data | Temporal width (data points) | 49, 98 |
| parameter | Spectral width (Mel frequencys) | 64, 128, 256 |

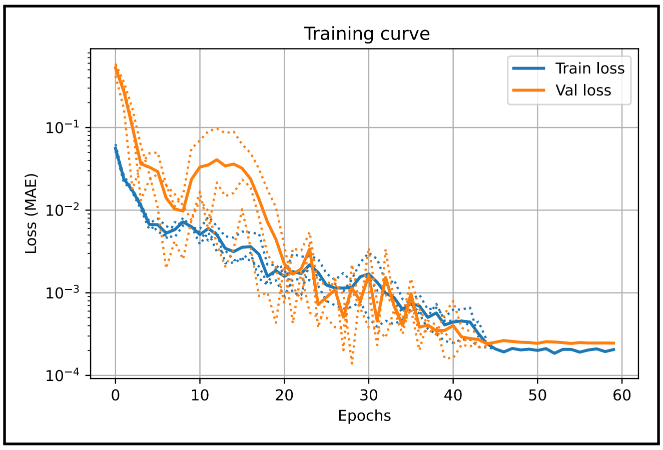

**Figure 5.** The 5-Fold validation curve for the best parameters found.

## 5. RESULTS

This section presents the results of the baseline model and the transfer learning. The task is to detect the first occurrence of an anomaly, typically associated with the onset of degradation. However, the tests are run until a stop condition is reached, and there is no way to determine when this occurs. A subset of bearing tests was manually inspected, and experts labeled the time at which degradation starts based on the vibration magnitude. There are nine bearings with available timestamps. In addition, a final report is available for each bearing where the final conditions and type of failure are specified.

The first evaluation concerns the time difference between the predicted first anomaly and the reported label. In general, it is a good sign if the model detects anomalies before the time labeled by the experts as this would point towards above expert knowledge capabilities. However, this can easily be obtained by reporting anomalies since the beginning of the test, which would be evident false positives. There is no exact way of determining this time limit for the current experiments. Therefore, these values need to be assessed with some reservations. Equation (1) shows the formula for the percentage difference ($perc_{error}$), given the anomaly detection time ($t_d$), the expert's reference ($t_r$) and the total length of the experiment ($T_t$). Here, positive values correspond to detections ahead of time, and negative values correspond to delayed detections.

$$perc_{error} = 100x(t_d - t_r)/T_t. \tag{1}$$

The second evaluation concerns the state at the end of the test, which can be either damaged or healthy, and is based on the end test report. For this evaluation, the accuracy, precision, and recall are reported. The formulas for accuracy, precision, and recall are shown in Eqs. (2), (3), and (4), respectively. In these cases, the true positives are anomalies correctly detected, and the abbreviations are as follows: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}; \tag{2}$$

**Table 7.** Baseline results, showing the difference between the anomaly detection and the reference timestamp.

| Bearing | Total duration (s) | Reference (s) | First detection (s) | Diff. | % Diff. | Abs % diff. |
|---|---|---|---|---|---|---|
| A83 | 7 930 | 7 100 | 7 090 | 10 | 0.14 | 0.14 |
| A84 | 35 538 | 33 478 | 28 608 | 4 870 | 14.55 | 14.55 |

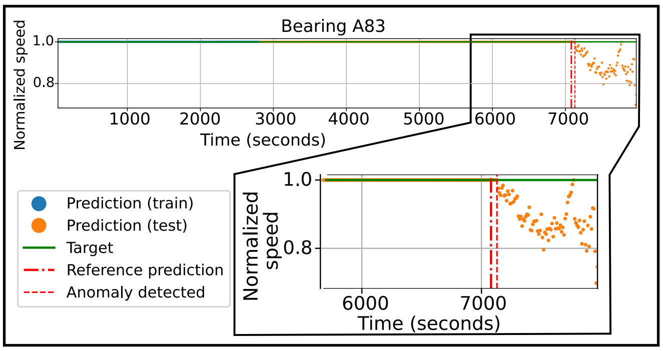

**Figure 6.** Normalized speed predictions for indented bearing A83. The anomaly is correctly detected in the proximity of the reference timestamp.

$$Precision = \frac{TP}{TP + FP}; \tag{3}$$

$$Recall = \frac{TP}{TP + FN}. \tag{4}$$

## 5.1. Baseline Model Performance

Table 7 presents the results for the first anomaly detected against the expert's reference. For the two bearings with available labels, A83 and A84, the anomalies are detected with a delay of 50 s and ahead by 4870 s (81 min), respectively. Figures 6 and 7 show the results for bearing A83, and Figs. 8 and 9 for bearing A84. The labeled anomaly corresponds to the sudden increase in the prediction error. The detection threshold was not tuned as this would require observing the error rates of the bearings after the training phase, which would defeat the purpose of making a fast-transferable approach. Notice that the missing data points for bearing A84 correspond to periods in which the machine was stopped.

Table 8 summarizes the results of the final bearing condition evaluation. The obtained accuracy is 98%, with only one out of 18 healthy bearings being classified as abnormal, namely a precision of (P=94.4%), a perfect recall (R=1), and no false negatives (FN=0). Figure 10 shows the target and predictions for an unindented bearing that does not fail during the test. Figure 11 shows the predictions against the selected threshold; the prediction error never surpasses the threshold. In summary, this indicates that the baseline model has an excellent sensibility for detecting anomalies and that, in no instance, it causes false positives.

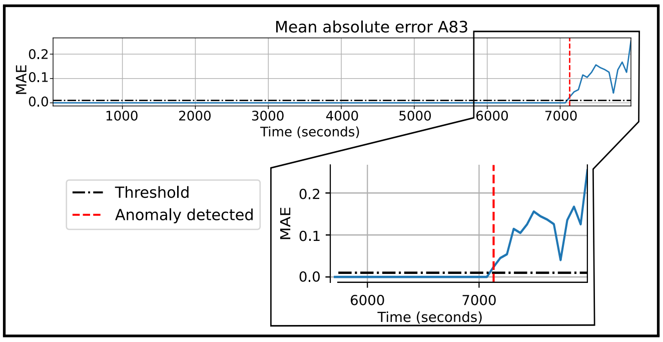

**Figure 7.** Mean error and first reported anomaly for indented bearing A83. The anomaly is reported when the prediction error surpasses the threshold.

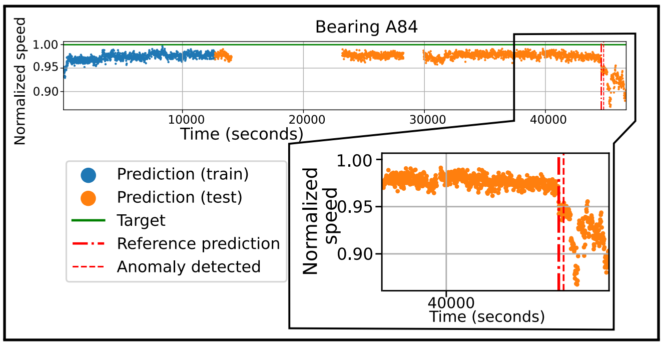

**Figure 8.** Normalized speed predictions for indented bearing A84. The anomaly is correctly detected in the proximity of the reference timestamp.

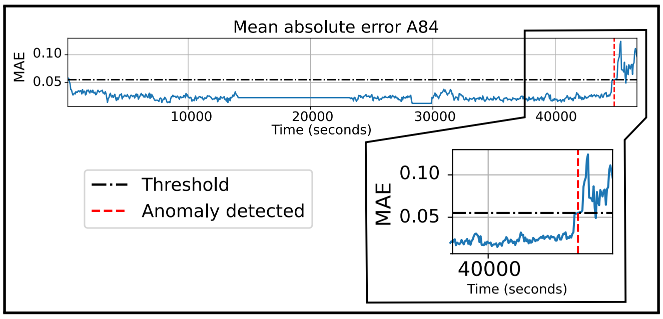

**Figure 9.** Mean error and first reported anomaly for indented bearing A84. The anomaly is reported when the prediction error surpasses the threshold.

## 5.2. Transfer Learning Performance

After demonstrating the effectiveness of the baseline model in identifying vibration patterns to diagnose the bearing condition, the TL, by fine-tuning is evaluated. Table 9 presents the results after training for the validation data, corresponding to the test's beginning for three hold-out bearings. The results indicate how well the model is adapted to the target data over previously unseen data. The TL advantage is evident as the error estimation is one order of magnitude smaller than the baseline reference. Interestingly, the errors obtained for the model that only uses the target data are more significant than the baseline. This fact indicates an underfit scenario, most likely due to insufficient training data. The expected result is that the baseline model will be more likely to predict false positives, whereas the model trained with only the target data will be unable to detect anomalies.

The transfer learning models can detect anomalies ahead of time or with a slight delay concerning the reference value. Table 10 summarizes the results and compares them against the baseline, which is the model without fine-tuning. The results

**Table 8.** Confusion matrix of the baseline model.

| | | Predicted | |
|---|---|---|---|
| | | Anomaly | Healthy |
| Actual | Anomaly | 100% | 0% |
| | Healthy | 2.22% | 97.77% |

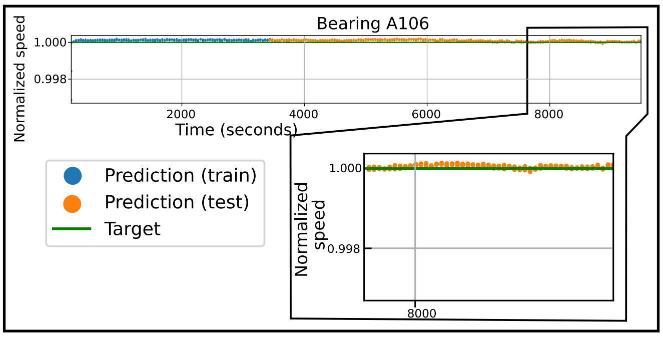

**Figure 10.** Normalized speed predictions for unindented bearing A106.

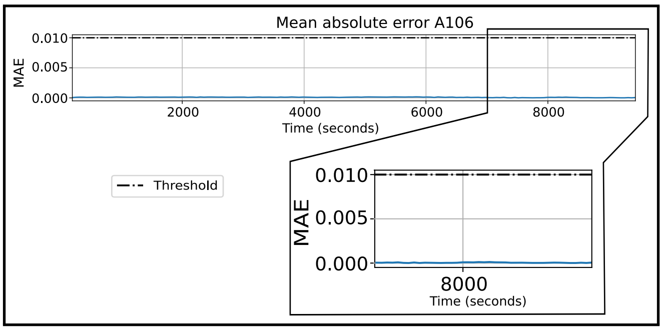

**Figure 11.** Mean error for unindented bearing A106. The threshold is not surpassed, hence no anomaly is reported.

from the baseline trained using only the target data are not presented as these models failed to detect the anomaly in all cases. Notably, the baseline model tends to predict anomalies considerably ahead of the reference, attributed to significant prediction errors due to the new operating condition being unseen during the model's training. Whether this is a false positive or better performance is challenging to assess as bearings are only visually inspected at the end of the test. The models trained using only the training target data underfit and failed to detect anomalies due to insufficient target speed data.

Concerning the result, both the baseline model and the adapted model can detect anomalies in all 14 indented bearings before the end of the test (This corresponds to an accuracy of 100% and a precision of 100%. Note, however, that TN and FN are not defined as no healthy bearings are evaluated in the transferred models). This fact indicates the robustness and effectiveness of the models in identifying anomalies. However, the validation results presented in Table 9 demonstrate the superiority of the TL approach. TL models have smaller validation errors for the target conditions, and their predictions are closer to the expert's reference. In summary, the TL models outperform both the baseline and target-only models. This highlights the value of leveraging the knowledge learned from the abundant source domain to improve the anomaly detection performance in the target domains.

**Table 9.** Validation results (mean absolute error) for each model over all the folds for the validation data.

| Target speed | Baseline | Only Target | Transfer learning |
|---|---|---|---|
| 1100 | 0.188 ± 0.016 | 0.179 ± 0.196 | 0.028 ± 0.015 |
| 1200 | 0.137 ± 0.076 | 0.485 ± 0.066 | 0.013 ± 0.003 |
| 1300 | 0.208 ± 0.069 | 0.439 ± 0.199 | 0.013 ± 0.002 |
| 1400 | 0.071 ± 0.015 | 0.451 ± 0.214 | 0.019 ± 0.005 |
| 1500 | 0.096 ± 0.019 | 0.498 ± 0.156 | 0.019 ± 0.008 |
| 1600 | 0.190 ± 0.100 | 0.706 ± 0.014 | 0.021 ± 0.005 |
| 1700 | 0.129 ± 0.080 | 0.636 ± 0.097 | 0.021 ± 0.005 |
| 1800 | 0.222 ± 0.190 | 0.706 ± 0.126 | 0.015 ± 0.004 |
| 1900 | 0.245 ± 0.081 | 0.886 ± 0.110 | 0.015 ± 0.006 |
| Mean | 0.165 | 0.554 | 0.018 |

## 5.3. Discussion

In contrast to many other studies in the field where models are evaluated based on their ability to classify faulty and healthy samples accurately, this research approached the challenge of online anomaly detection by precisely determining the first anomaly's occurrence time. Therefore, rather than focusing on accuracy or precision, the performance focused on the time delay or overhead of detecting a single anomaly per bearing. This emphasis was chosen due to the critical nature of detecting the initial anomaly in numerous real-world applications, as it often signifies machinery failure or marks the onset of degradation in the context of bearings. Extra anomalies detected afterward can also be informative, but they may be trivial to notice or be reported too late to be useful for decision-making. In addition, a practical consideration when deploying monitoring solutions is the reactive actions that must be taken. To obtain more robust results, one could also consider training models to optimize the precision or recall of the models based on the cost of decision-making, where the false positives and false negatives incur specific expenses, and the end goal is to reduce their respective risks. For example, as shown in Table 10, the baseline models are incurring considerable false positives. In contrast, the transferred models incur false negatives shortly after detection (less than 6 minutes) in three cases. Expert knowledge can be used to select the most convenient trade-off between both metrics.

The transferal challenge investigated concerns scenarios with limited data in the target domain. The presented TL model uses a small fraction of data, which, combined with a small architecture, allows training a model within minutes. This means the models can be effectively adapted and deployed to monitor a new bearing. Notice that only the first 35% samples are used under a highly imbalanced transferal scenario. Due to the limited number of available labeled bearings, data from within a run was used to train the baseline and the transferal models. As previously mentioned, this can lead to a small degree of information leakage due to samples from the training and validation being temporally correlated with the test data.

Further reducing the train margin (below 35%) could make the evaluation more rigorous. Notice that results were also presented for hold-out tests to demonstrate that the generalization is valid over unseen bearings, where no information leakage occurs. As more data becomes available, a more detailed evaluation can be done to measure correct generalization or a more rigorous assessment using repeated cross-validation. Overall, this shows how the presented methodology can be trained and adapted easily in real settings, as it would only require a few minutes of incoming data to be adapted to previously unseen conditions.

An important caveat to address concerns the model selection approach. As explained in Section 4.4, the best model

**Table 10.** Results for transfer learning. The reference corresponds to the annotations made by experts. Positive differences correspond to detections ahead of the reference whereas negative differences correspond to delayed detections.

| | | | Baseline | | | Transfer learning | | |
|---|---|---|---|---|---|---|---|---|
| Bearing | Total duration | Reference | First detection | Difference | Abs Pct Diff | First detection | Difference | Abs Pct Diff |
| A146 | 28261 | 28031 | 11540 | 16491 | 58.35 | 15120 | 12911 | 45.68 |
| A148 | 8670 | 6760 | 7400 | -640 | 7.38 | 7100 | -340 | 3.92 |
| A150 | 12370 | 10150 | 6000 | 4150 | 33.55 | 4630 | 5520 | 44.62 |
| A151 | 18320 | 18280 | 8940 | 9340 | 50.98 | 11890 | 6390 | 34.88 |
| A154 | 7680 | 5940 | 4620 | 1320 | 17.19 | 5990 | -50 | 0.65 |
| A155 | 23051 | 20681 | 8580 | 12101 | 52.50 | 20951 | -270 | 1.17 |
| A156 | 44412 | 42162 | 25741 | 16421 | 36.97 | 28251 | 13911 | 31.32 |
| A158 | 19821 | 19821 | 7260 | 12561 | 63.37 | 10630 | 9191 | 46.37 |
| Mean | | | | | 38.19 | | | 26.08 |

is selected based on the mean absolute error of predicting the operational speed. This optimization procedure does not necessarily correspond to finding the best model for anomaly detection. Neither does it imply the reported error rate is more reliable. The reported error is not a parameterized value and cannot be compared across models. In other words, we can only rely on the changes in the prediction error, but the magnitudes reported by different models are not comparable as these are not distance metrics. In addition, research has empirically shown that models trained to perform better on a given task may lead to underperforming transferable models.[68] At this point, this is still an open question that is still being researched in the field, and that should be considered in future research.

Finally, the presented work used the minimum and maximum speeds for the baseline models. However, other speed combinations are possible. The current example demonstrates a case of interpolation, where the new speeds are within the seen data. Extrapolation scenarios where the speeds are outside of the range can also be addressed following this methodology. However, in Table 2, the sample size for tests with a speed different than 2000 rpm is considerably smaller.

## 6. FUTURE WORK

After this study, some future research tracks are proposed as follows:

1. Basis function decompositions. One of the crucial steps in PdM tasks is the preferred data representations. This research chose the mel spectrogram due to its compact representation. However, other spectral models could be considered and evaluated. For example, wavelet and chirplet transforms can generate multi-resolution representations in frequency and time, which could benefit from richer features and allow anomaly detection at a more fine-grained time resolution. Furthermore, different families of basis functions can enhance the expected patterns. In this case, the vibration patterns of healthy bearings can be described by wavelets such as Daubechies and Morlet, which can be improved using these basis functions. Recent works have shown ways of learning basis functions and their parameters via deep learning[69] in the context of condition monitoring.

2. Data augmentation. Early work in this research used data augmentation techniques from SpecAugment,[70] frequency masking, and time masking to improve the validation error of the baseline model. This proved to be effective. However, the obtained robustness was such that the model could correctly identify the operational speed even if the signal was considerably distorted due to degradation. Data augmentation has proven extremely useful in training deep learning models and should also be considered in the context of transfer learning. Some methods that should be considered include data transformations[6] and generative approaches.[71] Furthermore, modifying the objective function of the presented baseline model could allow the SpecAugment to be used, for example, by finding structures using self-supervised learning instead of supervised learning.

3. Transfer learning for specific scenarios. Transfer learning has gained much attention in more specialized methods in recent years. This work presented a classical finetuning approach, which has become a common standard due to its ease of deployment. However, it is only suited for supervised scenarios in which a sufficient amount of labeled data exists. More recent approaches that could be investigated and are relevant for industrial applications include adversarial learning for self-supervised or unsupervised transfer[72], few-shot learning for anomaly detection for scenarios where few labels are available in the target domain[73], and the integration of explainability for TL-AD.[74]

4. Advanced architectures. Finally, more recent lightweight architectures could be considered, such as MobileNetV3[64] and EfficientNet[75]. However, as discussed in this research, efficiency in a pre-trained task does not guarantee the best results for transfer learning.

## 7. CONCLUSION

This study presented the successful implementation of a transfer learning approach to adapt a neural network for bearing diagnosis under new operating conditions. The transferability was evaluated on a selected number of runs whose vibration signals were carefully inspected, and the bearings' outcomes were assessed after the tests. The objective was the identification of anomalies in an unsupervised way, meaning no labels were provided for these events.

The results showed that the baseline model trained only on the source data has a high accuracy for anomaly detection in the source domain (accuracy 98% and precision 94.4%) and that the adapted models derived from it can learn the new operating conditions with limited samples. It was found that the baseline model trained with the source domain and the adapted model could detect all the anomalies in the target domain (accuracy 100% and precision 100%). In contrast, the model trained using the target domain data had insufficient information and was underfitted, causing it to be unable to detect any of the anomalies (0% accuracy).

When comparing the detection precision, it was found that the adapted models provided predictions more in line with the experts' knowledge. An absolute percentage difference of 26.08% was obtained for the adapted models and 38.19% for the baseline model. The baseline model consistently predicted the first anomaly considerably ahead of time. Detecting anomalies too early could imply a risk of them being actual false positives. We hypothesize this is caused by the domain shift and the baseline models mistaking different operational conditions as anomalies.

The results showed the advantage of using transfer learning due to two key benefits: allowing the reuse of previously collected data and modifying the model to perform under new conditions. It was demonstrated that using only the target domain data was insufficient for detecting anomalies, and using only the source domain data yielded models that were too sensible to the operating condition changes. These findings support the adoption of transfer learning techniques for bearing condition monitoring, enabling earlier detection of anomalies and allowing the reuse of previously collected data.

## ACKNOWLEDGEMENTS

The authors would like to thank Bram Robberechts for performing the tests within the Smart Maintenance Living Lab project. *Funding* This work was supported by the Flemish Government under the “Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen” programme and the VLAIO subsidies for research.

## CONFLICT OF INTEREST

The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## REFERENCES

[1] Ooijevaar, T. H., Pichler, K., Di, Y., Devos, S., Volckaert, B., Van Hoecke, S., and Hesch, C. Smart machine maintenance enabled by a condition monitoring living lab. In *IFAC-PapersOnLine*, volume 52, 376–381. Elsevier B.V., Amsterdam, The Netherlands, 2019. ISSN 24058963. https://doi.org/10.1016/j.ifacol.2019.11.704.

[2] Heng, A., Zhang, S., Tan, A. C., and Mathew, J. Rotating machinery prognostics: State of the art, challenges and opportunities. *Mechanical Systems and Signal Processing*, **23**(3), 724–739, 2009. ISSN 08883270. https://doi.org/10.1016/j.ymssp.2008.06.009.

[3] Jardine, A. K., Lin, D., and Banjevic, D. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing*, **20**(7), 1483–1510, 2006. ISSN 08883270. https://doi.org/10.1016/j.ymssp.2005.09.012.

[4] Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems (NIPS)*, **27**, 2014.

[5] Zhang, R., Peng, Z., Wu, L., Yao, B., and Guan, Y. Fault diagnosis from raw sensor data using deep neural networks considering temporal coherence. *Sensors (Switzerland)*, **17**(3), 2017. ISSN 14248220. https://doi.org/10.3390/s17030549.

[6] Kong, Y., Qin, Z., Han, Q., Wang, T., and Chu, F. Enhanced dictionary learning based sparse classification approach with applications to planetary bearing fault diagnosis. *Applied Acoustics*, **196**, 108870, 2022. ISSN 0003-682X. https://doi.org/10.1016/j.apacoust.2022.108870.

[7] Gebraeel, N., Lawley, M., Liu, R., and Parmeshwaran, V. Residual life predictions from vibration-based degradation signals: A neural network approach. *IEEE Transactions on Industrial Electronics*, **51**(3), 694–700, 2004. ISSN 02780046. https://doi.org/10.1109/TIE.2004.824875.

[8] Huang, R., Xi, L., Li, X., Richard Liu, C., Qiu, H., and Lee, J. Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods. *Mechanical Systems and Signal Processing*, **21**(1), 193–207, 2007. ISSN 08883270. https://doi.org/10.1016/j.ymssp.2005.11.008.

[9] Serradilla, O., Zugasti, E., Rodriguez, J., and Zurutuza, U. Deep learning models for predictive maintenance: a survey, comparison, challenges and prospects. *Applied Intelligence*, **52**(10), 10934–10964, 2022. ISSN 1573-7497. https://doi.org/10.1007/s10489-021-03004-y.

[10] Guo, L., Lei, Y., Xing, S., Yan, T., and Li, N. Deep Convolutional Transfer Learning Network: A New Method for Intelligent Fault Diagnosis of Machines with Unlabeled Data. *IEEE Transactions on Industrial Electronics*, **66**(9), 7316–7325, 2019. ISSN 02780046. https://doi.org/10.1109/TIE.2018.2877090.

[11] Liu, L., Wang, L., and Yu, Z. Remaining Useful Life Estimation of Aircraft Engines Based on Deep Convolution Neural Network and LightGBM Combination Model. *International Journal of Computational Intelligence Systems*, **14**(1), 1–10, 2021. ISSN 18756883. https://doi.org/10.1007/S44196-021-00020-1/TABLES/4.

[12] Xia, M., Li, T., Xu, L., Liu, L., and De Silva, C. W. Fault Diagnosis for Rotating Machinery Using Multiple Sensors and Convolutional Neural Networks. *IEEE/ASME Transactions on Mechatronics*, **23**(1), 101–110, 2018. ISSN 10834435. https://doi.org/10.1109/TMECH.2017.2728371.

[13] Eren, L., Ince, T., and Kiranyaz, S. A Generic Intelligent Bearing Fault Diagnosis System Using Compact Adaptive 1D CNN Classifier. *Journal of Signal Processing Systems*, **91**(2), 179–189, 2019. ISSN 19398115. https://doi.org/10.1007/s11265-018-1378-3.

[14] Sateesh Babu, G., Zhao, P., and Li, X. L. Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life. *Database Systems for Advanced Applications*, 214–228, 2016.

[15] Wen, L., Li, X., Gao, L., and Zhang, Y. A New Convolutional Neural Network-Based Data-Driven Fault Diagnosis Method. *IEEE Transactions on Industrial Electronics*, **65**(7), 5990–5998, 2018. ISSN 02780046. https://doi.org/10.1109/TIE.2017.2774777.

[16] Hemmer, M., Van Khang, H., Robbersmyr, K., Waag, T., and Meyer, T. Fault Classification of Axial and Radial Roller Bearings Using Transfer Learning through a Pretrained Convolutional Neural Network. *Designs*, **2**(4), 56, 2018. ISSN 2411-9660. https://doi.org/10.3390/designs2040056.

[17] Li, S., Liu, G., Tang, X., Lu, J., and Hu, J. An Ensemble Deep Convolutional Neural Network Model with Improved D-S Evidence Fusion for Bearing Fault Diagnosis. *Sensors*, **17**(8), 1729, 2017. ISSN 1424-8220. https://doi.org/10.3390/s17081729.

[18] Xu, G., Liu, M., Jiang, Z., Söffker, D., and Shen, W. Bearing Fault Diagnosis Method Based on Deep Convolutional Neural Network and Random Forest Ensemble Learning. *Sensors*, **19**(5), 1088, 2019. ISSN 1424-8220. https://doi.org/10.3390/s19051088.

[19] Chen, R., Huang, X., Yang, L., Xu, X., Zhang, X., and Zhang, Y. Intelligent fault diagnosis method of planetary gearboxes based on convolution neural network and discrete wavelet transform. *Computers in Industry*, **106**, 48–59, 2019. ISSN 0166-3615. https://doi.org/10.1016/J.COMPIND.2018.11.003.

[20] Yoo, Y. and Baek, J. G. A Novel Image Feature for the Remaining Useful Lifetime Prediction of Bearings Based on Continuous Wavelet Transform and Convolutional Neural Network. *Applied Sciences*, **8**(7), 1102, 2018. ISSN 2076-3417. https://doi.org/10.3390/app8071102.

[21] Zhang, K., Tang, B., Deng, L., and Liu, X. A hybrid attention improved ResNet based fault diagnosis method of wind turbines gearbox. *Measurement*, **179**, 109491, 2021. ISSN 0263-2241. https://doi.org/10.1016/J.MEASUREMENT.2021.109491.

[22] Zhao, M., Kang, M., Tang, B., and Pecht, M. Deep Residual Networks with Dynamically Weighted Wavelet Coefficients for Fault Diagnosis of Planetary Gearboxes. *IEEE Transactions on Industrial Electronics*, **65**(5), 4290–4300, 2018. ISSN 02780046. https://doi.org/10.1109/TIE.2017.2762639.

[23] Kraus, M. and Feuerriegel, S. Forecasting remaining useful life: Interpretable deep learning approach via variational Bayesian inferences. *Decision Support Systems*, **125**, 113100, 2019. ISSN 01679236. https://doi.org/10.1016/j.dss.2019.113100.

[24] Lee, J., Lee, Y. C., and Kim, J. T. Fault detection based on one-class deep learning for manufacturing applications limited to an imbalanced database. *Journal of Manufacturing Systems*, **57**, 357–366, 2020. ISSN 0278-6125. https://doi.org/10.1016/J.JMSY.2020.10.013.

[25] Li, M., Yu, D., Chen, Z., Xiahou, K., Ji, T., and Wu, Q. H. A Data-Driven Residual-Based Method for Fault Diagnosis and Isolation in Wind Turbines. *IEEE Transactions on Sustainable Energy*, **10**(2), 895–904, 2019. ISSN 19493029. https://doi.org/10.1109/TSTE.2018.2853990.

[26] Wang, F., Liu, X., Deng, G., Yu, X., Li, H., and Han, Q. Remaining Life Prediction Method for Rolling Bearing Based on the Long Short-Term Memory Network. *Neural Processing Letters*, **50**(3), 2437–2454, 2019. ISSN 1573773X. https://doi.org/10.1007/s11063-019-10016-w.

[27] Zhu, Y., Zhu, C., Tan, J., Tan, Y., and Rao, L. Anomaly detection and condition monitoring of wind turbine gearbox based on LSTM-FS and transfer learning. *Renewable Energy*, **189**, 90–103, 2022. ISSN 0960-1481. https://doi.org/10.1016/J.RENENE.2022.02.061.

[28] Wu, C., Feng, F., Wu, S., Jiang, P., and Wang, J. A method for constructing rolling bearing lifetime health indicator based on multi-scale convolutional neural networks. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, **41**(11), 526, 2019. ISSN 18063691. https://doi.org/10.1007/s40430-019-2010-6.

[29] Pinedo-Sánchez, L. A., Mercado Ravell, D. A., and Carballo Monsivais, C. A. Vibration analysis in bearings for failure prevention using CNN. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, **42**(12), 628, 2020. ISSN 18063691. https://doi.org/10.1007/s40430-020-02711-w.

[30] Marei, M., Zaatari, S. E., and Li, W. Transfer learning enabled convolutional neural networks for estimating health state of cutting tools. *Robotics and Computer-Integrated Manufacturing*, **71**, 102145, 2021. ISSN 0736-5845. https://doi.org/10.1016/J.RCIM.2021.102145.

[31] Ying, Z., Shu, L., Kizaki, T., Iwama, M., and Sugita, N. Hybrid Approach for Onsite Monitoring and Anomaly Detection of Cutting Tool Life. *Procedia CIRP*, **104**, 1541–1546, 2021. ISSN 2212-8271. https://doi.org/10.1016/J.PROCIR.2021.11.260.

[32] Zhang, K., Tang, B., Deng, L., Tan, Q., and Yu, H. A fault diagnosis method for wind turbines gearbox based on adaptive loss weighted meta-ResNet under noisy labels. *Mechanical Systems and Signal Processing*, **161**, 107963, 2021. ISSN 0888-3270. https://doi.org/10.1016/J.YMSSP.2021.107963.

[33] Guo, L., Li, N., Jia, F., Lei, Y., and Lin, J. A recurrent neural network based health indicator for remaining useful life prediction of bearings. *Neurocomputing*, **240**, 98–109, 2017. ISSN 18728286. https://doi.org/10.1016/j.neucom.2017.02.045.

[34] He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, **25**, 770–778, 2016. ISSN 10636919. https://doi.org/10.1109/CVPR.2016.90.

[35] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the Inception Architecture for Computer Vision. 2818–2826, 2016. ISSN 10636919. https://doi.org/10.1109/CVPR.2016.308.

[36] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. 2009. https://doi.org/10.1109/CVPR.2009.5206848.

[37] Hong, F., Song, J., Meng, H., Rui, W., Fang, F., and Guangming, Z. A novel framework on intelligent detection for module defects of PV plant combining the visible and infrared images. *Solar Energy*, **236**, 406–416, 2022. ISSN 0038-092X. https://doi.org/10.1016/J.SOLENER.2022.03.018.

[38] Lu, W., Liang, B., Cheng, Y., Meng, D., Member, S., and Yang, J. Deep Model Based Domain Adaptation for Fault Diagnosis. *IEEE Transactions on Industrial Electronics*, **64**(3), 2296–2305, 2017.

[39] Hu, Q., Si, X., Qin, A., Lv, Y., and Liu, M. Balanced Adaptation Regularization Based Transfer Learning for Unsupervised Cross-Domain Fault Diagnosis. *IEEE Sensors Journal*, **22**(12), 12139–12151, 2022. ISSN 15581748. https://doi.org/10.1109/JSEN.2022.3174396.

[40] Michau, G. and Fink, O. Unsupervised transfer learning for anomaly detection: Application to complementary operating condition transfer. *Knowledge-Based Systems*, **216**, 106816, 2021. ISSN 0950-7051. https://doi.org/10.1016/J.KNOSYS.2021.106816.

[41] Rao, W., Qu, Y., Gao, L., Sun, X., Wu, Y., and Zhang, B. Transferable network with Siamese architecture for anomaly detection in hyperspectral images. *International Journal of Applied Earth Observation and Geoinformation*, **106**, 102669, 2022. ISSN 1569-8432. https://doi.org/10.1016/J.JAG.2021.102669.

[42] Saenko, K., Kulis, B., Fritz, M., and Darrell, T. Adapting visual category models to new domains. In C.V.E. 2010, editor, *Lecture Notes in Computer Science*, volume 6314, 213–226. Springer Verlag, 2010. ISBN 364215560X. ISSN 16113349. https://doi.org/10.1007/978-3-642-15561-1_16/.

[43] Ramakrishnan, R., Nagabandi, B., Eusebio, J., Chakraborty, S., Venkateswara, H., and Panchanathan, S. Deep Hashing Network for Unsupervised Domain Adaptation. *Domain Adaptation in Computer Vision with Deep Learning*, 57–74, 2020. https://doi.org/10.1007/978-3-030-45529-3_4.

[44] Li, D., Yang, Y., Song, Y. Z., and Hospedales, T. M. Deeper, Broader and Artier Domain Generalization. *2017 IEEE International Conference on Computer Vision (ICCV)*, 5543–5551, 2017.

[45] Nectoux, P., Gouriveau, R., Medjaher, K., Ramasso, E., Chebel-Morello, B., Zerhouni, N., and Varnier, C. PRONOSTIA: An experimental platform for bearings accelerated degradation tests. In *Conference on Prognostics and Health Management.*, 1–8. 2012.

[46] Lee, J., Qiu, H., Yu, G., Lin, J., and Rexnord Technical Services. *IMS, University of Cincinnati. "Bearing Data Set"*. NASA Ames Research Center, Moffett Field, CA, 2007.

[47] Nieves Avendano, D., Vandermoortele, N., Soete, C., Moens, P., Ompusunggu, A. P., Deschrijver, D., and Van Hoecke, S. A Semi-Supervised Approach with Monotonic Constraints for Improved Remaining Useful Life Estimation. *Sensors 2022, Vol. 22, Page 1590*, **22**(4), 1590, 2022. ISSN 1424-8220. https://doi.org/10.3390/S22041590.

[48] Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, **86**(11), 2278–2324, 1998. https://doi.org/10.1109/5.726791.

[49] Karen, S. and Andrew, Z. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2015.

[50] Hasanpour, S. H., Rouhani, M., Fayyaz, M., and Sabokrou, M. Lets keep it simple, Using simple architectures to outperform deeper and more complex architectures. *arXiv preprint arXiv:1608.06037*, 2016.

[51] Han, S., Pool, J., Tran, J., and Dally, W. J. Learning Both Weights and Connections for Efficient Neural Networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, volume 1 of *NIPS'15*, 1135–1143. MIT Press, Cambridge, MA, USA, 2015.

[52] Li, H., Kadav, A., Durdanovic, I., Samet, H., and Graf, H. P. Pruning Filters for Efficient ConvNets. *arXiv preprint arXiv:1608.08710*, 2017.

[53] Chenzhuo, Z., Song, H., Huizi, M., and William J., D. Trained Ternary Quantization. 2017.

[54] Wang, C. H., Huang, K. Y., Yao, Y., Chen, J. C., Shuai, H. H., and Cheng, W. H. Lightweight Deep Learning: An Overview. *IEEE Consumer Electronics Magazine*, 1–12, 2022. https://doi.org/10.1109/MCE.2022.3181759.

[55] Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. *arXiv:1602.07360*, 2016.

[56] Barret, Z., Vijay, V., Jonathon, S., and Quoc V., L. Learning Transferable Architectures for Scalable Image Recognition. 2017.

[57] Howard, A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. 2017.

[58] Dong, N., Zhang, C., and Chen, H. Research on Fault Diagnosis Method of Rolling Bearing Based on MobileNet V2. In H. Zhang, Y. Ji, T. Liu, X. Sun, and A.D. Ball, editors, *Proceedings of TEPEN 2022*, 528–533. Springer Nature Switzerland, Cham, 2023. ISBN 978-3-031-26193-0.

[59] Yu, W. and Lv, P. An End-to-End Intelligent Fault Diagnosis Application for Rolling Bearing Based on MobileNet. *IEEE Access*, **9**, 41925–41933, 2021. https://doi.org/10.1109/ACCESS.2021.3065195.

[60] Pham, M. T., Kim, J. M., and Kim, C. H. Deep Learning-Based Bearing Fault Diagnosis Method for Embedded Systems. *Sensors*, **20**(23), 2020. ISSN 1424-8220. https://doi.org/10.3390/s20236886.

[61] Guzmán-Torres, J. A., Morales-Rosales, L. A., Algredo-Badillo, I., Tinoco-Guerrero, G., Lobato-Báez, M., and Melchor-Barriga, J. O. Deep learning techniques for multi-class classification of asphalt damage based on hamburg-wheel tracking test results. *Case Studies in Construction Materials*, **19**, e02378, 2023. ISSN 2214-5095. https://doi.org/10.1016/j.cscm.2023.e02378.

[62] Kim, B., Yuvaraj, N., Park, H. W., Preethaa, K. S., Pandian, R. A., and Lee, D. E. Investigation of steel frame damage based on computer vision and deep learning. *Automation in Construction*, **132**, 103941, 2021. ISSN 0926-5805. https://doi.org/10.1016/j.autcon.2021.103941.

[63] Bianco, S., Cadène, R., Celona, L., and Napoletano, P. Benchmark Analysis of Representative Deep Neural Network Architectures. *IEEE Access*, **6**, 64270–64277, 2018.

[64] Andrew, H., Mark, S., Grace, C., Liang-Chieh, C., Bo, C., Mingxing, T., Weijun, W., Yukun, Z., Ruoming, P., Vijay, V., Quoc V., L., and Hartwig, A. Searching for MobileNetV3. *arXiv:1905.02244*, 2019.

[65] Alfredo, C., Adam, P., and Eugenio, C. An Analysis of Deep Neural Network Models for Practical Applications. *arXiv:1605.07678*, 2016.

[66] Moens, P., Bracke, V., Soete, C., Vanden Hautte, S., Nieves Avendano, D., Ooijevaar, T., Devos, S., Volckaert, B., and Van Hoecke, S. Scalable Fleet Monitoring and Visualization for Smart Machine Maintenance and Industrial IoT Applications. *Sensors*, **20**(15), 4308, 2020. ISSN 1424-8220. https://doi.org/10.3390/s20154308.

[67] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L. C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 4510–4520, 2018. ISSN 10636919. https://doi.org/10.1109/CVPR.2018.00474.

[68] Kornblith, S., Chen, T., Lee, H., and Norouzi, M. Why Do Better Loss Functions Lead to Less Transferable Features? *Advances in Neural Information Processing Systems*, **34**, 28648–28662, 2021.

[69] Michau, G., Frusque, G., and Fink, O. Fully learnable deep wavelet transform for unsupervised monitoring of high-frequency time series. *Proceedings of the National Academy of Sciences*, **119**, e2106598119, 2022. https://doi.org/10.1073/pnas.2106598119.

[70] Park, D. S., Chan, W., Zhang, Y., Chiu, C. C., Zoph, B., Cubuk, E. D., and Le, Q. V. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. *Interspeech 2019*, 2019. https://doi.org/10.21437/interspeech.2019-2680.

[71] Ruan, D., Chen, X., Gühmann, C., and Yan, J. Improvement of Generative Adversarial Network and Its Application in Bearing Fault Diagnosis: A Review. *Lubricants*, **11**(2), 2023. ISSN 2075-4442. https://doi.org/10.3390/lubricants11020074.

[72] Kimura, D., Chaudhury, S., Narita, M., Munawar, A., and Tachibana, R. Adversarial Discriminative Attention for Robust Anomaly Detection. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2161–2170. 2020. https://doi.org/10.1109/WACV45572.2020.9093428.

[73] Belton, N., Hagos, M. T., Lawlor, A., and Curran, K. M. FewSOME: One-Class Few Shot Anomaly Detection With Siamese Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2977–2986. 2023.

[74] Serradilla, O., Zugasti, E., Ramirez de Okariz, J., Rodriguez, J., and Zurutuza, U. Adaptable and Explainable Predictive Maintenance: Semi-Supervised Deep Learning for Anomaly Detection and Diagnosis in Press Machine Data. *Applied Sciences*, **11**(16), 2021. ISSN 2076-3417. https://doi.org/10.3390/app11167376.

[75] Mingxing, T. and Quoc V., L. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv:1905.11946*, 2019.