machines

MDPI

Article

Vibration-Based Anomaly Detection in Industrial Machines: A Comparison of Autoencoders and Latent Spaces

Luca Radicioni *, Francesco Morgan Bono and Simone Cinquemani

Department of Mechanical Engineering, Politecnico di Milano, 20133 Milan, Italy; francescomorgan.bono@polimi.it (F.M.B.); simone.cinquemani@polimi.it (S.C.)
* Correspondence: luca.radicioni@polimi.it

Abstract: In industrial settings, machinery components inevitably wear and degrade due to friction between moving parts. To address this, various maintenance strategies, including corrective, preventive, and predictive maintenance, are commonly employed. This paper focuses on predictive maintenance through vibration analysis, utilizing data-driven models. This study explores the application of unsupervised learning methods, particularly Convolutional Autoencoders (CAEs) and variational Autoencoders (VAEs), for anomaly detection (AD) in vibration signals. By transforming vibration signals into images using the Synchrosqueezing Transform (SST), this research leverages the strengths of convolutional neural networks (CNNs) in image processing, which have proven effective in AD, especially at the pixel level. The methodology involves training CAEs and VAEs on data from machinery in healthy condition and testing them on new data samples representing different levels of system degradation. The results indicate that models with spatial latent spaces outperform those with dense latent spaces in terms of reconstruction accuracy and AD capabilities. However, VAEs did not yield satisfactory results, likely because reconstruction-based metrics are not entirely useful for AD purposes in such models. This study also highlights the potential of ReLU residuals in enhancing the visibility of anomalies. The data used in this study are openly available.

Keywords: predictive maintenance; system degradation; anomaly detection; autoencoders; variational autoencoders; synchrosqueezing; vibration analysis

check for updates

Academic Editors: Michael I. Friswell and Zoltan-Iosif Korka

Received: 15 January 2025
Revised: 8 February 2025
Accepted: 10 February 2025
Published: 12 February 2025

Citation: Radicioni, L.; Bono, F.M.; Cinquemani, S. Vibration-Based Anomaly Detection in Industrial Machines: A Comparison of Autoencoders and Latent Spaces. *Machines* **2025**, *13*, 139. https://doi.org/10.3390/machines13020139

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Regardless of the specific field an industry operates in, when moving elements are involved, there will inevitably be wear and degradation of surfaces in contact. By definition, machines have moving components, which means that some of their parts must be replaced to restore nominal operating conditions. Consequently, industrial plant managers are constantly dealing with the maintenance of their equipment to maintain high availability and limit downtimes. In industry, maintenance strategies are regularly classified into three areas: *corrective*, *preventive*, and *predictive* [1]. In corrective maintenance, replacements are made only after a failure has occurred, which means that the consequent equipment unavailability is accepted in favor of low servicing and monitoring costs. In preventive maintenance, inspections and actions are performed based on predefined schedules. The scheduling can be time-based or condition-based [2]. Predictive maintenance occurs when maintenance is performed just before a fault happens, thereby reducing both planned and unplanned downtime. Continuous monitoring of the equipment improves the latter, which necessarily requires the adoption of a sensor mesh.

The literature has widely focused on preventive maintenance in recent decades due to its attractiveness in minimizing downtime costs. Rytter [3] and later Worden [4] performed a remarkable job in formalizing the problem of fault identification; the resulting hierarchical framework prescribes the possible level of accuracy achievable from the monitoring system. Moreover, two main areas of research are addressed in these works: the construction of an optimal sensor network and the consequent data processing required to create a suitable predictive model. In this work, the sensors considered are those capable of measuring the vibration of a machine, such as accelerometers. The methods used to create the predictive models are data-driven.

The friction between two surfaces in contact produces vibrations in the bodies involved. Although it has been widely demonstrated how vibrations convey relevant information about the nature of a process [5,6], modeling the friction process is highly time-consuming and strictly depends on the observed system. Vibration condition monitoring is a topic that has been extensively addressed in the literature. Due to the extent of this field and the numerous traditional approaches, a comprehensive discussion is unfeasible here. Interested readers may find valuable information in [7], whereas a practical tutorial can be found in [8]. However, it is still worth noting that a rigorous modeling of the phenomenon in novel scenarios could be such a complex task [9] that it might appear unfeasible to construct and validate a vibration-based model for most practical cases. For this reason, the adoption of data-driven modeling is extremely profitable in this context: a data-driven model can be fitted onto data with virtually no human supervision. Data-driven modeling is generally referred to as "machine learning" in many contexts. In learning theory, *supervised* refers to methodologies that can be applied to a dataset where each instance is accompanied by its corresponding target. Conversely, *unsupervised* indicates a problem where the dataset is unlabeled. While supervised algorithms are applied to create predictive models, unsupervised ones are used to identify structure within the dataset without the need for the "labeling" task, which is carried out by humans and is thus both time-consuming and resource-intensive. This property of unsupervised algorithms makes them suitable for various contexts, ranging from identifying patterns and clusters in big data to reducing the dimensionality of virtual objects, a feature frequently employed in the fields of image and signal compression [10].

A significant number of publications in the field of industrial monitoring deal with *supervised* learning [11], although the application of such models in real-life scenarios is often impractical due to the difficulty of mapping all possible failure modes. Instead, unsupervised learning methods are appealing since the models can be fitted onto the system in a "nominal" condition and then identify if the system deviates from the nominal condition. To support this concept, Eltouny et al. [12] proposed a systematic review focused on applications in vibration-based condition monitoring using unsupervised methods, highlighting the strengths of these approaches: they are particularly praised for not requiring the collection of damaged data, not relying on physics-based models, and addressing the difficulties in collecting datasets from laboratory samples or field tests.

Unsupervised learning methods for novelty detection have seen rapid development, especially in image processing. Novelty identification in graphical patterns has progressed significantly at the pixel level, particularly due to the adoption of convolutional neural networks. In this field, a major contribution was made by Wang et al. [13], who introduced an innovative similarity index to quantify differences in images. Another significant contribution is attributed to Selvaraju et al. [14], who proposed a method (Grad-CAM) to highlight the regions of an image that contributed the most to the outcome of a convolutional neural network. But more importantly, anomaly detection (AD) in images has experienced significant improvements in recent decade thanks to the introduction of

Convolutional Autoencoders (CAEs) and the more refined *variational Autoencoders* (VAEs), classes of neural networks explicitly trained to reconstruct the input object [15]. The use of Autoencoders (AEs) in AD at the pixel level has been addressed by Baur et al. [16], who compared the performances of AE–GAN and VAE–GAN with both dense and spatial latent spaces to unsupervisedly identify brain lesions. Bergmann et al. [17] introduced a dataset (MVTec AD) specifically designed to test unsupervised methods for images, demonstrating the performance of the most common neural network-based methods. Venkataramanan et al. [18] proposed an attention-guided framework (CAVGA) to improve the AD capabilities of three unsupervised datasets. Gangloff et al. [19] proposed a variation of VAE (VAE-GRF), introducing a novel anomaly map. Finally, Nguyen et al. [20] published a comparison of state-of-the-art VAE methods.

In the context of vibration condition monitoring, deep learning [21], and, specifically, AEs, have proven effective in identifying structural anomalies [22]. Moreover, having assessed the capabilities of AEs to identify anomalies in images, it is worth exploring the concept of transforming vibration signals into "images". This approach has been extensively investigated in the literature, particularly using the Short-Time Fourier Transform (STFT) or Discrete Wavelet Transform (DWT). Recently, some research has focused on a variation of these techniques called the *Synchrosqueezing Transform* (SST), which produces spectrograms with clearer and sharper components. The literature on the use of the SST for AD or condition monitoring is relatively sparse, although some notable examples exist. Wang et al. [23] utilized the SST to overcome the limitations of Fourier transform and wavelet analysis in seismic data analysis. Hazra et al. employed the SST to study anomalies involving gearboxes before using unsupervised methods in several applications [24–26]. Mao [27] used a variation of the SST, called the adaptive SST, to identify oil reservoirs. Tian et al. [28] applied the SST to detect ferromagnetic objects. Finally, Rasul et al. [29] exploited the SST before feeding a deep learning model built for classification. Generally speaking, the SST can be applied to both the STFT and the DWT. However, in this work, we explore the synchrosqueezing of a Short-Time Fourier Transform (FSST). This choice is based on the observation that the frequency content of the STFT is often immediately understandable, as well as the meaning of the main frequency components, which can be directly used in mathematical relationships accounting for periodic functions. The FSST is essentially a refinement of the STFT, where sparsity is promoted, making the main frequency components clearer. In contrast, the interpretation of the DWT is not as straightforward. Although DWTs have proven to be highly effective in AD, in this phase it has been decided to postpone their exploration to future works.

From this comprehensive review on AD with AEs and SST, the following observations have been derived:

- In the field of AD based on vibrations, using the SST instead of the STFT and DWT appears to be a profitable strategy due to the sparsity and sharpness of the resulting spectrograms.
- The use of convolutional neural networks (CNNs) seems to be the main reason behind the outstanding performance of unsupervised AD in images, which also allows for good interpretability, especially as evidenced by the heatmaps produced by Grad-CAM [14].
- Many studies employ AEs with a spatial latent space. It is observed that models with this latent space generally feature better performance than those with dense layers, with fewer weights.
- The performance of VAEs is generally superior to that of AEs, at the cost of a slight complication, i.e., the implementation of the KL loss in the latent space. At the same

time, more refined VAE variations do improve overall performance, although this increase is not always justified by the complexity of those models.

All these observations suggest that mapping raw vibrations into a domain where they can be treated as images allows for analysis with a suitable CAE.

Main Contributions

The objective of this paper is to investigate the application of image processing techniques for vibration monitoring. This involves transforming the signals from the time domain to the time–frequency domain using an SST. Subsequently, various types of CAEs are trained on initial samples of acceleration data collected from industrial machinery in healthy condition. Finally, these models are tested on new data samples, which represent different levels of system degradation. The main contributions of the paper are listed as follows:

- The SST is systematically applied to an extensive dataset produced by an accelerometer mounted on machinery subjected to degradation, so as to enable a vibration-based unsupervised AD methodology.
- A comparison of two types of unsupervised AD models is conducted: CAEs and VAEs. Additionally, for each model, two latent spaces are examined: the common dense latent space and the less common spatial (convolutional) latent space. This comparison enables an understanding of the practical implications of this combinations in terms of AD performance, model structure, and best practices in model training.
- The introduction of "ReLU residual", a metric to enhance the visibility of anomalies at pixel level.

This paper is arranged as follows: In Section 2, the main theoretical concepts are briefly introduced. Section 3 describes the structure of the dataset and the methodology adopted to compare the different versions of AEs. In Section 4, the results are shown and discussed. Finally, in Section 5, some conclusive comments are drawn. The data used in this study are available at https://bit.ly/4ahDnzY (accessed on 26 January 2025).

2. Theoretical Background

To achieve the objectives proposed in the introduction, the theoretical background of the main concepts addressed in this study is presented here.

2.1. Unsupervised Anomaly Detection

Anomaly detection (AD), also known as *outlier* or *novelty* detection, focuses on methods that use data to identify observations that significantly deviate from the concept of "nominality". The methods commonly used in AD are *unsupervised*, meaning that anomalous observations are unlabeled, and when labels do exist, they are generally insufficient to effectively characterize the concept of anomalousness [30]. In the context of industrial monitoring, a simple yet effective scheme to build an unsupervised monitoring solution is depicted in Figure 1. The unsupervised model is trained on data from the system in a "healthy" condition. It is hypothesized that machinery is in good condition at the beginning of its service life. The baseline (training set) is therefore selected from a period during which the machinery is operating in an optimal state. This procedure is similar to that used in unsupervised anomaly detection with images, where the training set typically consists of photos of objects with no superficial defects. However, for such datasets, there is usually an anomaly map at the pixel level, which allows for evaluating the model's performance during deployment for each image. Ideally, the model should identify the patterns associated with the anomaly map as anomalous. Finally, the trained model is then deployed on new data from the monitored system. The use of suitable metrics should help the machinery's responsible personnel make informed decisions on when to perform maintenance.

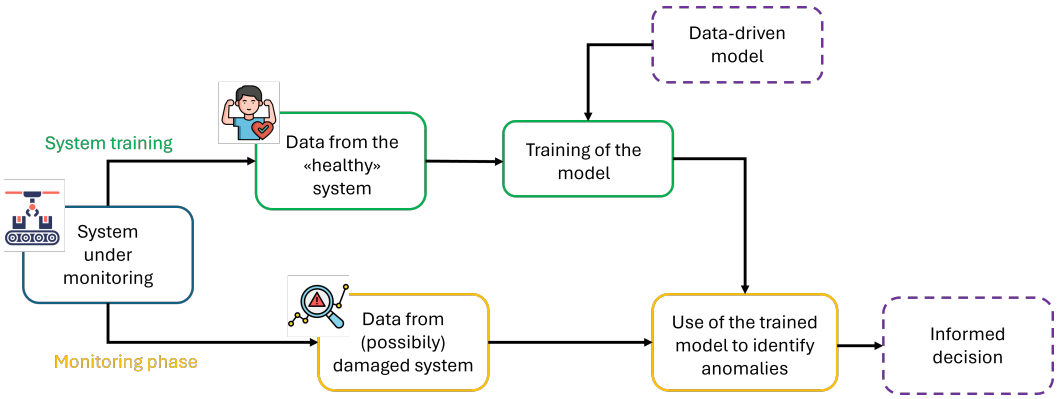


Figure 1. Scheme that represents the training and monitoring phases for a generic industrial process, based on an unsupervised AD approach. The dashed boxes represent the only tasks that require human supervision.

2.2. Time or Space Analysis?

Acceleration signals are ergodic and time-dependent, meaning they can be viewed as collections of values (vectors) with strong statistical–temporal relationships among samples. However, the commonly used neural network layers designed to capture temporal relationships, such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), are notoriously difficult to train, leading to long training times and issues like vanishing gradients [31,32]. By transforming the time series into spatial matrices, we can leverage the power of convolutional layers, which are more efficient to train compared to recurrent layers. Convolutional layers operate on local regions of the input, allowing them to capture spatial patterns without the need for long-range dependencies. This is why it is increasingly common to find research where digital signals are transformed into *images* through time–frequency analysis, using techniques such as the STFT and DWT.

2.3. Synchrosqueezing Transform

The Synchrosqueezing Transform (SST) was initially introduced in the context of audio signal processing and is particularly appealing due to its properties of invertibility and sparsity [33]. It can be applied to both the Fourier Transform and the Wavelet Transform; in the former case, it is considered an enhancement of the STFT. The STFT of a signal $x(t)$ is expressed as follows:

$$\mathrm{STFT}_x(t,\omega) = \int_{-\infty}^{\infty} x(\tau) \cdot w(t-\tau) \cdot e^{-j\omega\tau}\, d\tau \tag{1}$$

where $w(t)$ is a window function centered at t, and j is the imaginary unit. While the STFT provides a time–frequency representation of the signal, it can suffer from blurring due to the Heisenberg uncertainty principle.

The SST aims to offer a more localized time–frequency representation by redistributing the energy of the STFT. This is achieved by computing a modified STFT, where each frequency component is "squeezed" to its central frequency, resulting in the SST of an STFT, usually indicated as the FSST. The FSST of $x(t)$ is defined as follows [34]:

$$\mathrm{FSST}_x(t,\omega) = \int_{-\infty}^{\infty} |\mathrm{STFT}_x(t,\omega')|^2 \cdot \delta(\omega - \omega_c(t,\omega'))\, d\omega' \tag{2}$$

where $\omega_c(t,\omega')$ is the central frequency at time t and frequency ω', and δ is the Dirac delta function. By redistributing the energy of the STFT in this way, the FSST provides a more accurate and interpretable time–frequency representation of the signal. This makes it particularly useful for analyzing non-stationary signals, where the spectral content varies

over time. In time series analysis, this can help uncover hidden structures and patterns in the data that are not evident in the raw time series or the simple STFT. In Figure 2, an example of the results by applying the STFT and the FSST to the same accelerometer signal is shown. Clearly, the SST promotes a more sparse representation of the signal, making the dominant frequency contents evident.

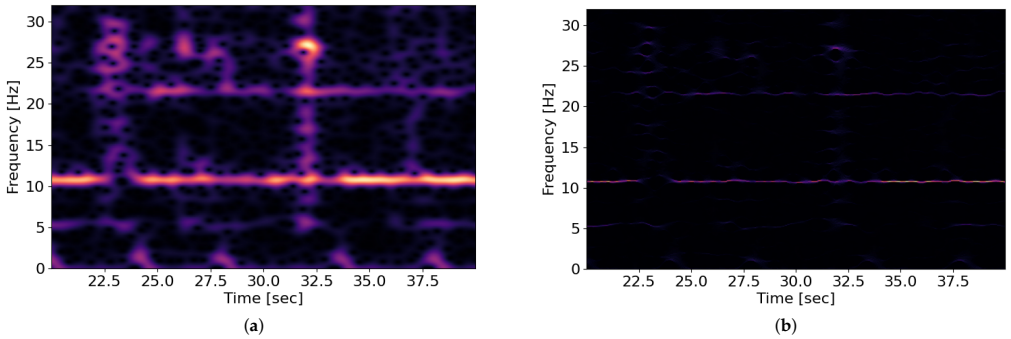


Figure 2. Comparison of the STFT (**a**) and the FSST (**b**), applied to the same piece of acceleration signal. The SST approach promotes sparsity in the time–frequency domain.

2.4. Autoencoders

Among the various methods for achieving AD, Autoencoders (AEs) fall into the category of reconstruction-based models. These models are based on the idea that if a model is trained to successfully reconstruct a certain category of objects, it will fail to reconstruct objects with characteristics significantly different from those in the "nominal" category. In practice, AEs are a type of neural network characterized by the following properties [30]: Consider a feature map $\phi_\theta : \mathcal{X} \to \mathcal{X}$, which maps the input data space $\mathcal{X}$ onto itself. Let us assume that ϕ_θ is obtained by concatenating two sub-functions, $\phi_\theta = (\phi_d \circ \phi_e)_\theta$, where $\phi_e : \mathcal{X} \to \mathcal{Z}$ is the "encoding" function, and $\phi_d : \mathcal{Z} \to \mathcal{X}$ is the "decoding" function. The training objective is to learn the set of parameters θ so that an unlabeled input is reconstructed with minimal error, i.e., $\phi_\theta(\mathbf{x}) = \phi_d(\phi_e(\mathbf{x})) = \hat{\mathbf{x}}$. If the loss function considered is the mean absolute error (MAE), the problem formalizes as follows:

$$\min_\theta \frac{1}{n} \sum_{i=1}^{n} |\mathbf{x}_i - (\phi_d \circ \phi_e)_\theta \ (\mathbf{x}_i)| + \mathcal{R} \tag{3}$$

where $\mathcal{R}$ refers to the regularization terms that might be introduced to promote certain characteristics in the model. An example of a simple AE is shown in Figure 3a.

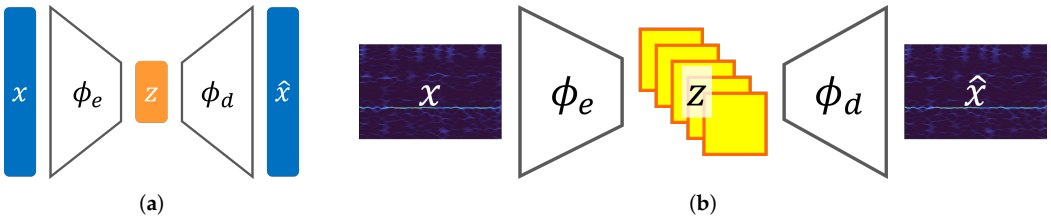


Figure 3. Graphical representation of a simple AE (**a**); the input element x is mapped into an n-dimensional latent space—z—and then mapped back into the original data space. Graphical representation of a CAE with a spatial latent space (**b**); the input image is elaborated through a fully convolutional architecture.

When dealing with objects characterized by spatial information, convolutional CAEs are the most common unsupervised models. Like an AE, the structure of a CAE is symmetric and composed of convolutional layers in both the encoder and decoder. During the encoding phase, the CAE utilizes a series of convolutional and pooling layers to compress

the input data into a lower-dimensional latent space. This process preserves the most critical features of the input while eliminating less important details and noise. In the decoding phase, the CAE takes this compressed representation and employs a sequence of upsampling and convolutional layers to reconstruct it back to the original input space. In CAEs, the latent space can be either dense—meaning there is a one-axis tensor in the latent space ($\mathbf{z} \in \mathbb{R}^h$)—or spatial, where the convolutional structure is preserved in the latent space ($\mathbf{z} \in \mathbb{R}^{h \times w \times c}$). The former configuration requires a flatten layer, which introduces many weights into the model, whereas the latter configuration ensures a model characterized by a reduced number of weights, thereby assuring better performance [16]. An example of a CAE with a spatial latent space is shown in Figure 3b.

2.5. Variational Autoencoders

One main drawback of AEs in their "vanilla" version is that they do not consider the structure of the latent space. Typically, the latent space does not produce an ordered representation, making AD in the latent space less effective [35]. However, the literature review in the introduction has shown that generative models have been extensively explored, even in the context of AD. Variational Autoencoders (VAEs) are a class of neural generative models that maximize data likelihood by optimizing the evidence lower bound (ELBO), defined in the generic variational approach as follows [36]:

$$\mathcal{L}_{\text{ELBO}} = \mathbb{E}_{z \sim Q(z|x)} \log(P(x|z)) - KL(Q(z|x) \,||\, P(z)) \tag{4}$$

The first term represents the expected value of the log-likelihood of the datapoint x given the latent variable z, with the expectation taken with respect to the approximate posterior distribution $Q(z|x)$. The second term represents the Kullback–Leibler divergence between the approximate posterior distribution $Q(z|x)$ and the prior distribution of the latent variable $P(z)$. This KL divergence encourages a well-structured latent space and prevents the training process from converging to trivial solutions.

Assuming that $Q(z|x)$ is normally distributed, with mean and variance given by the encoding functions $\mu_e(x)$ and $\sigma_e^2(x)$, and considering a standard normal distribution as the prior, the KL divergence can be computed in closed form:

$$KL(G(\mu_e(x), \sigma_e(x)), G(0,1)) = \frac{1}{2}\left(\mu_e(x)^2 + \sigma_e^2(x) - \log(\sigma_e^2(x)) - 1\right) \tag{5}$$

The Gaussian assumption also simplifies the first term, as it can be shown that

$$\mathbb{E}_{z \sim G(\mu_e(x),\, \sigma_e(x))} \log(P(x|z)) = (x - \phi_d(z))^2 \tag{6}$$

which is the reconstruction loss. This allows us to introduce the following compact equation for the ELBO problem in VAEs:

$$\mathcal{L}_{\text{ELBO}} = \mathcal{L}_{\text{R}} + \mathcal{L}_{\text{KL}} \tag{7}$$

$$\mathcal{L}_{\text{R}} = \mathbb{E}_{z \sim Q(z|x)} \log(P(x|z)) \tag{8}$$

$$\mathcal{L}_{\text{KL}} = -KL(Q(z|x) \,||\, P(z)) \tag{9}$$

where $\mathcal{L}_{\text{R}}$ is the reconstruction loss and $\mathcal{L}_{\text{KL}}$ is the KL divergence.

In VAEs, the encoding functions $\mu_e(x)$ and $\sigma_e^2(x)$, and the decoding function $\phi_d(z)$, are approximated by neural networks. Finally, backpropagation is enabled by the well-known reparameterization trick [37]. Examples of VAEs with dense and spatial latent spaces (sVAE) are shown in Figure 4a,b.

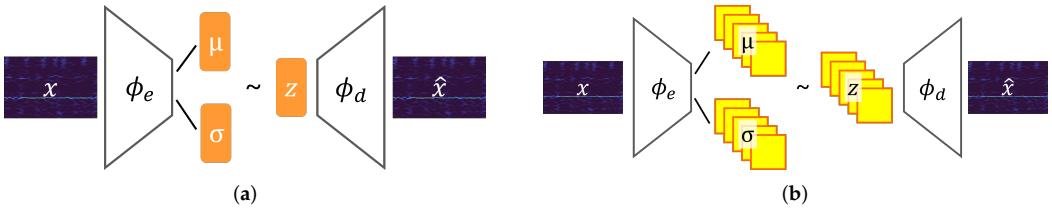


Figure 4. Example of a VAE with a dense latent space (**a**) and a spatial (sVAE) latent space (**b**).

2.6. Annealing in VAE Training

Training a VAE is notoriously challenging. Balancing the reconstruction error and the KL divergence has been the subject of numerous studies. A significant breakthrough in this field was achieved with the introduction of the β-VAE, which incorporates a weight that multiplies the KL divergence to help balance the two loss functions [38]. In this context, the VAE can be seen as a regularized version of the simple AE (refer to Equation (3)):

$$\mathcal{L}_\beta = \mathcal{L}_\mathrm{R} + \beta \cdot \mathcal{L}_\mathrm{KL} \tag{10}$$

Building on the β-VAE configuration, Hao et al. [39] proposed cyclical annealing of the KL divergence to mitigate KL vanishing, improving upon the concept of monotonic annealing [40]. In practice, the weight β increases after each epoch, starting from a null value. After a predefined number of epochs, β is reset to null, and the procedure repeats. An example of training following these procedures is shown in Figure 5.

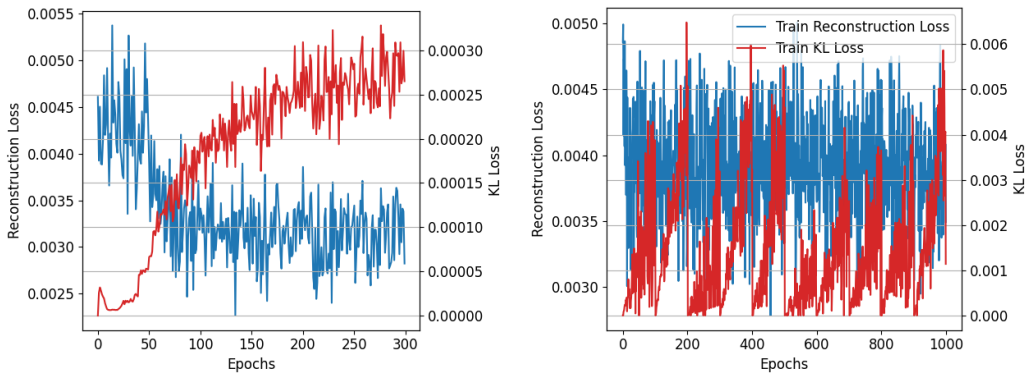


Figure 5. Example of a VAE's training process with different annealing strategies: monotonic annealing (**a**) and cycling annealing (**b**). The red curve refers to the product $\beta \cdot \mathcal{L}_\mathrm{KL}$.

3. Dataset Description and Methodology

The dataset considered in this research comes from machinery used for the transportation of lumped goods, specifically a conveyor chain. In this section, only the relevant information for the research scope is reported about the acquired data, introducing some simplifications on how the data were collected. However, a precise description of the system under study and the complete acquisition system is presented in [41], bearing in mind that this is an example of vibration monitoring of machinery subjected to degradation. Although the dataset used for testing data-driven models is commonly presented after the description of the methodology, here the opposite will be carried out to make the procedure prescribed in the methodology clearer for the reader.

3.1. Data Collection

An accelerometer (Figure 6) was mounted on the platform of a conveyor chain, where high degradation is expected for the material located between the platform and the moving chain. Several tests, each three hours long, were conducted, varying the boundary conditions: three main velocities were considered for the electric motor moving the chain, and three load configurations, varying the number of masses on the conveyor. The nine tests were then repeated for a total of four rounds to highlight the system's degradation after several working hours. From previous knowledge of the system, two main failure modes are expected to be visible in the measured accelerations due to the system's degradation:

1. The degradation of the chain elements produces an elongation of the average chain pitch. This will be reflected in the shifting of the main frequencies in the spectra towards lower-frequency components.
2. The degradation of the polyethylene sleeve between the platform and the chain generates an increase in the mean RMS of the vibrations, producing an increase in the frequency amplitudes.

To provide more evidence of these aspects, which are present but not visually clear in the spectra, three more rounds were generated, starting from the real data by artificially increasing the chain elongation, and then simulating a severe degradation of the chain elements. Further details on chain elongation and its effect in the frequency domain can be found in [41]. Additionally, artificially increasing the chain pitch by lowering the main frequency components is expected to facilitate the anomaly detection task performed by the models. This assertion will be corroborated in the results, and specifically in Section 4.2. The procedure followed to generate these artificial rounds is described in the next subsection.

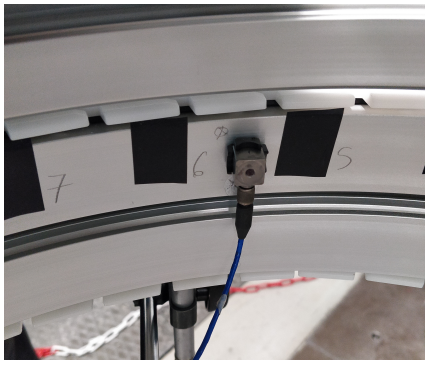


Figure 6. Photo of the piezoelectric accelerometer mounted on the platform of the conveyor (measuring axis oriented orthogonally to the platform surface).

3.2. Data Preprocessing

As discussed in depth in the introduction, the main purpose of this work is to leverage models suitable for image processing to achieve AD. This requires converting the vibration signals into images, i.e., 2-D tensors. It should be noted that most convolutional layers support the possibility of applying the convolutional operation separately to multiple channels, thus handling 3-D tensors. However, to enable a visual representation of the tensors and maintain consistency with the literature, only one channel (i.e., one accelerometer) is addressed here.

The FSST, as defined in Section 2.3, is applied to the accelerometer signals (1-D tensors) extracted from the tests, with time windows of 10 s. The resulting time–frequency spectrogram is then segmented into several 2-D tensors, each 20 s long. Considering a sampling frequency of 64 Hz, each tensor has a shape of 320×1280. Moreover, with 50 min of acceleration data for each test, this procedure yields 150 "images" per test, and approximately 1350 images per round.

An issue emerges: due to the different velocities at which tests are carried out, the main frequency contribution in the spectra changes significantly for different tests, even though they might occur within a few hours, with the system in practically the same condition. For this reason, a "normalization" of the spectrograms with respect to the motor velocity is applied by dividing the frequencies by the *theoretical frequency* (i.e., the frequency imposed by a chain with a nominal pitch for a given motor velocity) [35]. In this way, for the healthy system, the higher contributions in the spectrogram are expected at a normalized frequency of 1, and they shift towards lower normalized frequencies as the chain wears out. This feature is also exploited to create three more synthetic rounds by modifying the tensors from rounds 2, 3, and 4. This is carried out by considering theoretical link passage frequencies that are increased by 10%, 20%, and 30% compared to the real ones, respectively, so that the main dominant frequencies in the spectrogram are found at lower frequencies (i.e., the chain appears dangerously elongated). A summary of the complete dataset employed is schematically shown in Figure 7.

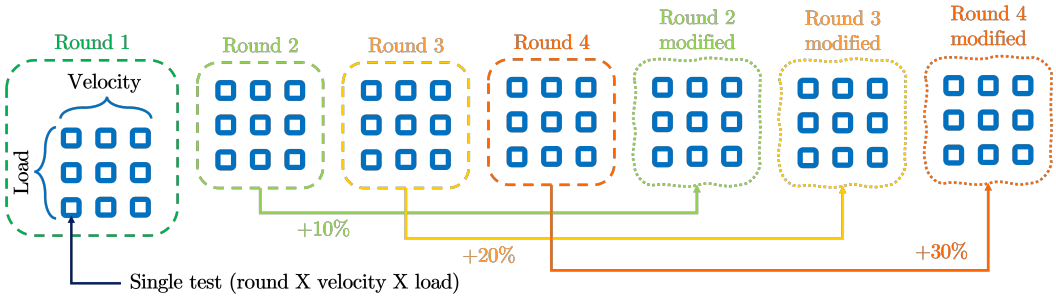


Figure 7. Visual representation of the dataset employed. The accelerations have been collected throughout 4 "rounds" for the machinery. Three more rounds have been generated synthetically.

Finally, to maintain the constant shape of the 2-D tensors, the normalized frequency axis is linearly interpolated onto a predefined normalized frequency vector, ranging from 0 to 2.5, with steps of 0.01 (251 elements). This technique, widely used in image processing [42], ensures consistent frequency indexing and enables the effective feeding of the tensors to the same neural network. The result of motor velocity normalization is represented in Figure 8. In this way, the load condition is the only external factor whose value the model does not have access to.

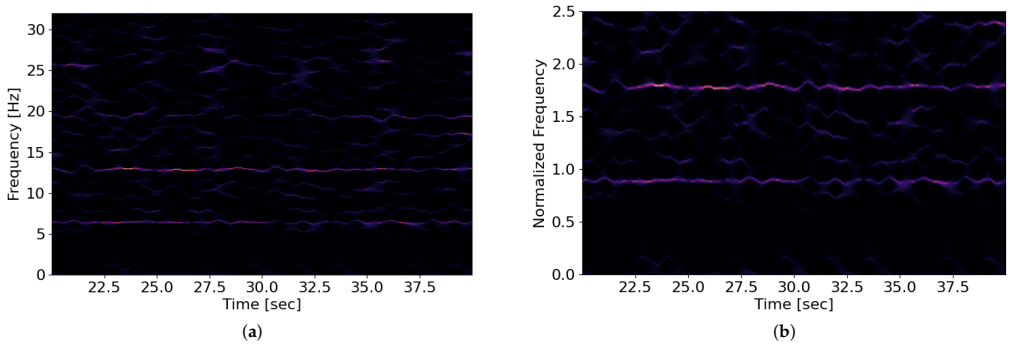


Figure 8. Motor velocity normalization. Starting from the SST spectrogram (**a**), the frequency axis is normalized with respect to the motor velocity. Then, a linear interpolation of the frequency axis to a predefine normalized frequency vector is performed (**b**).

3.3. Methodology

In health monitoring problems, and more generally in an AD setting, unsupervised approaches are currently the only profitable class of algorithms [30]. With reference to Figure 1, two key phases are clearly present in the pipeline:

- In the *training phase*, data representative of the machinery in good condition are used to train a data-driven model. The selection of the model is crucial, as the system's performance will be closely related to the model and its hyperparameters. In this work, four AEs are compared, each of which will be detailed in the next subsection. The spectrograms from round 1 serve as the training set used to fit the models.
- In the *monitoring phase*, the trained model is deployed to identify possible anomalies in the new data. To do so, several metrics must be introduced. In this work, for AD at the image level, the metrics introduced are the KL divergence (for the VAEs), the MAE between the input and the output tensor, and the ReLU MAE (for both the AE and VAE). At the pixel level, anomalies are identified by the point-wise difference between the input and output tensors (residual tensor) and the ReLU residual. All these metrics are detailed in the next subsection. All rounds will be evaluated in this phase to establish how the metrics change round after round.

The procedure follows this pipeline: the models are trained on round 1 and then tested on all rounds, evaluating the metrics.

3.4. Anomaly Metrics

To establish if an object is anomalous or not, it is crucial to define the *anomaly metrics*. In particular, two classes of metrics are introduced here: those at the image level, which allow us to infer if a spectrogram is anomalous or not, and those at the pixel level, which allow us to determine which elements of the spectrogram are anomalous.

At the pixel level, the metrics chosen are directly derived from the reconstruction loss: the *residual* spectrogram and the *ReLU residual* (Rectified Linear Unit) spectrogram. The residual spectrogram is the simple difference between the input tensor and the reconstructed tensor (with reference to the symbols introduced in Equation (3)):

$$\mathbf{r}_i = \mathbf{x}_i - (\phi_d \circ \phi_e)_\theta \ (\mathbf{x}_i) \tag{11}$$

where the pedix i refers to the i-th pixel. The ReLU residual spectrogram is developed from the observation that we are interested in what is present in the input and not in the output, and not vice versa, which is similar to what is carried out in Grad-CAM [14]:

$$\mathbf{r}_{\mathrm{ReLU},i} = \max(0, \mathbf{x}_i - (\phi_d \circ \phi_e)_\theta \ (\mathbf{x}_i)) \tag{12}$$

The ReLU MAE is expected to increase in correspondence with the pixels characterized by patterns that are not recognized by the AE.

Regarding the metrics at the image level, two metrics are used for the CAEs and three for the VAEs. One is the *simple MAE*, already presented in Equation (3). The other is the *ReLU MAE*, introduced to extend the idea of the ReLU residual from the pixel level to the image level:

$$\mathrm{MAE}_i = \frac{1}{n} \sum_{i=1}^{n} |\max(0, \ \mathbf{x}_i - (\phi_d \circ \phi_e)_\theta \ (\mathbf{x}_i))| \tag{13}$$

For the VAEs, the last metric is the KL divergence (Equation (5)).

3.5. Models

Four data-driven models are considered in this work; all share the same input and output dimensions (251×1280) and most of the layers. They all have the structure of a CAE. All the models have an encoder made up of four *2-D convolutional* layers, each followed by a *2-D Max pooling* layer. The decoder is made up of four *2-D Convolutional Transpose* layers, each followed by a *2-D Upsampling* layer. Finally, a further *2-D Convolutional Transpose* and a *2-D Cropping* layer are employed to adjust the output tensor dimension to the input one.

The common structure for the AEs is depicted in Figure 9. The four models differ in what happens in the *latent* space. The models are as follows:

1. A CAE with a *dense* latent space of 128 elements. To achieve this, a *flatten* layer is required. This model is characterized by 42,448,769 weights. The metric used in training is the MAE.
2. A CAE with a *spatial* latent space of dimensions (8, 40, 16). This model is characterized by 233,505 weights. The metric used in training is the MAE.
3. A VAE with a *dense* latent space of 128 elements. Similar to model 1, a *flatten* layer is required. This model is characterized by 21,379,969 weights. In training, the reconstruction error is the MAE, whereas annealing is applied to the KL divergence to regulate the β value from Equation (10).
4. A VAE with a *spatial* latent space of dimensions (8, 40, 16). This model is characterized by 134,048 weights. As per model 3, the reconstruction error is the MAE and annealing is applied to the KL divergence.

For the VAEs, both cycling and monotonic annealing will be tested, with different cycle length and maximum values for β.

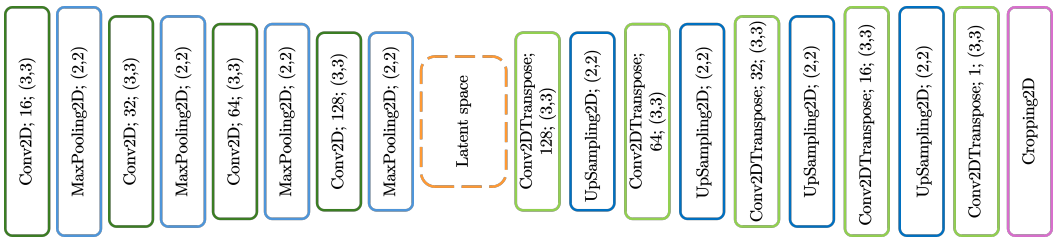


Figure 9. Common structure shared by all the four models evaluated in this work. For 2-D convolutional layers, the first number refers to the number of filters, the value between parenthesis to the kernel size.

4. Results

All the models were trained on a machine equipped with an NVIDIA GeForce RTX 3090 and using the Python library TensorFlow 2.12. In the first round, which serves as the training set, 15% of the spectrograms were kept for validation. The batch size chosen was eight, resulting in 120 batches per epoch. All the models were trained for 1000 epochs.

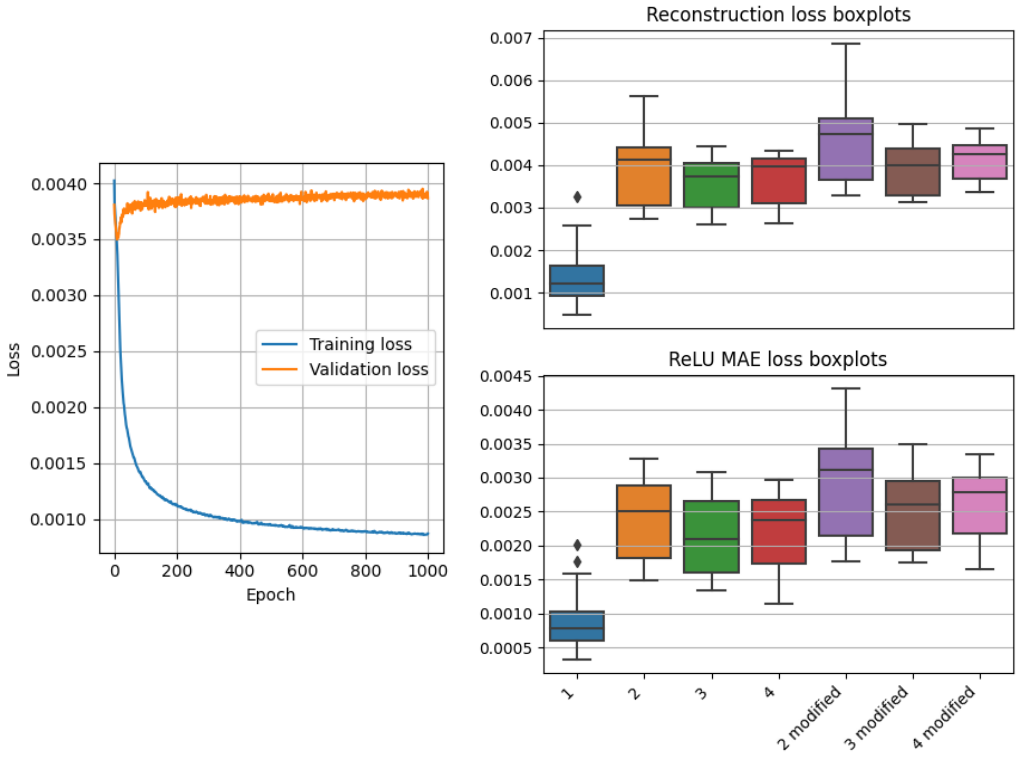
4.1. CAE with Dense Latent Space

The training time for this model was approximately 5 s per epoch. Figure 10a shows the training process, highlighting how the system began overfitting on the validation set after fewer than 100 epochs of training.

The trained model was then deployed on the spectrograms from all the rounds. Figure 10b presents the distributions of the metrics at the image level, that is, the MAE and the ReLU MAE, shown separately for each round. It is evident that both metrics generally increased between the training round (the first one) and the other rounds. However, the model seems unable to capture the progressive level of anomaly from each round, as all rounds share similar values for both metrics and do not feature a clear progression. Additionally, the two metrics yielded comparable results.

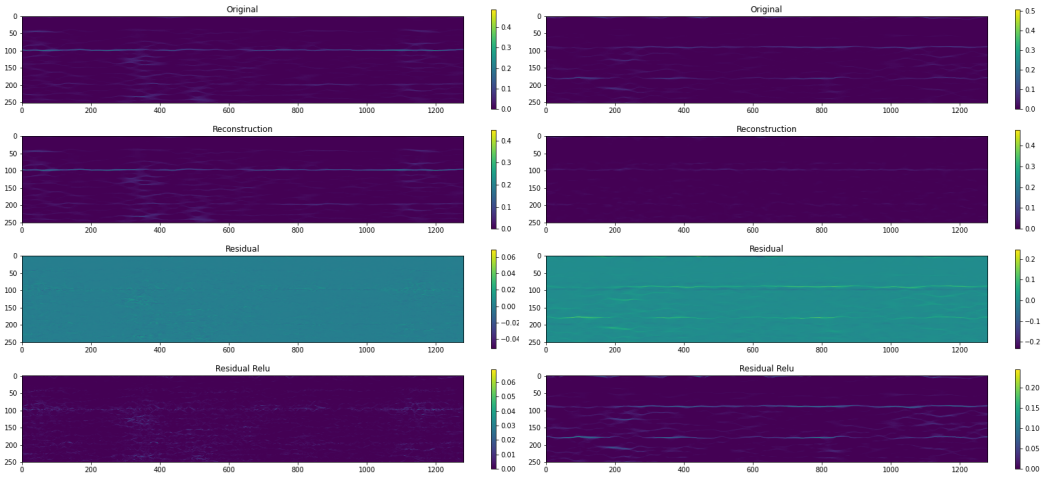
At the pixel level, it is possible to appreciate the anomalous frequency components. Two random tensors from the rounds that scored the minimum and maximum median values in Figure 10b were chosen, specifically round 1 and the modified round 2. Figure 11a shows an original spectrogram from round 1, its reconstructed spectrogram, the residual, and the ReLU residual, in this order. It is clear, especially from the ReLU residual, that most of the tensor was reconstructed correctly. Figure 11b presents the corresponding

results for a spectrogram from the modified round 2. Here, the model clearly identified the main two frequency components as anomalous, as they are still present in the residuals. Again, the ReLU residual shows more concise and understandable results compared to the simple residual.



(**a**) (**b**)

Figure 10. Results of CAE with dense latente space: (**a**) train and validation loss histories; (**b**) boxplots representing the distributions of the reconstruction losses (MAE and ReLU MAE) for each round.

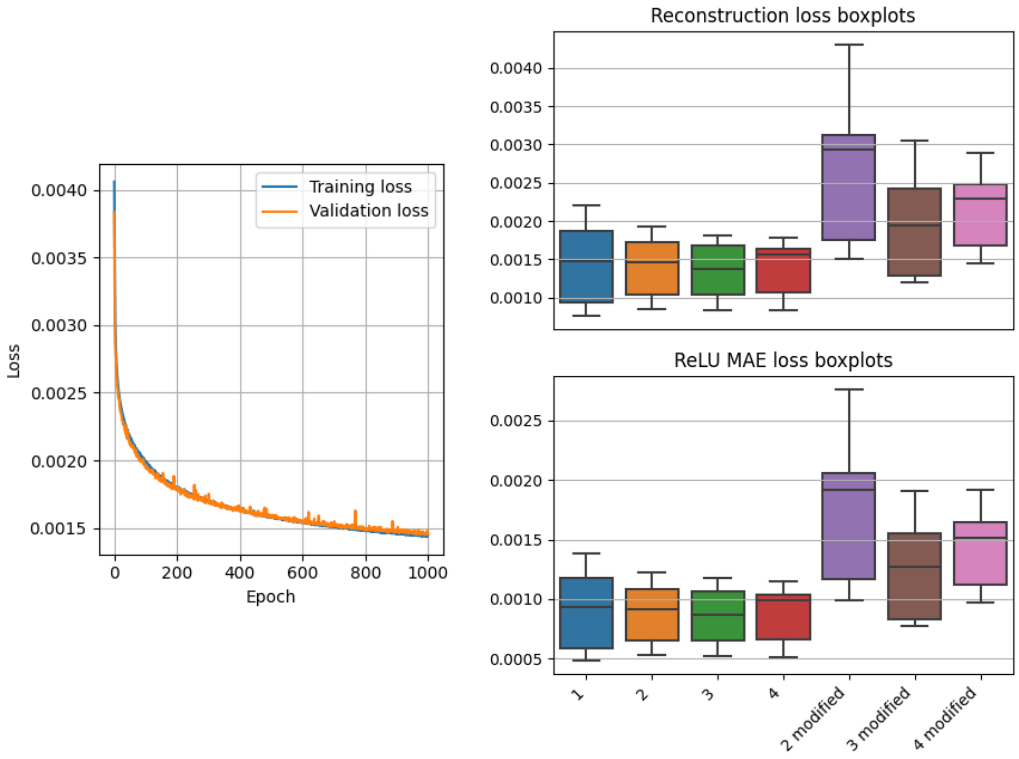


(**a**) (**b**)

Figure 11. Visualization of the results for the CAE with a dense latent space for a random spectrogram extracted from round 1 (**a**) and from round 2 modified (**b**). From top to bottom: original spectrogram, spectrogram reconstructed by the CAE, residual spectrogram, and ReLU residual.

4.2. CAE with Spatial Latent Space

The training time for this model was approximately 5 s per epoch. Figure 10a represents the training process for this model. From Figure 12a, it is evident that the spatial latent space achieves better performance compared to a dense latent space in terms of reconstruction accuracy. Moreover, after 1000 epochs, both training and validation losses are still improving.



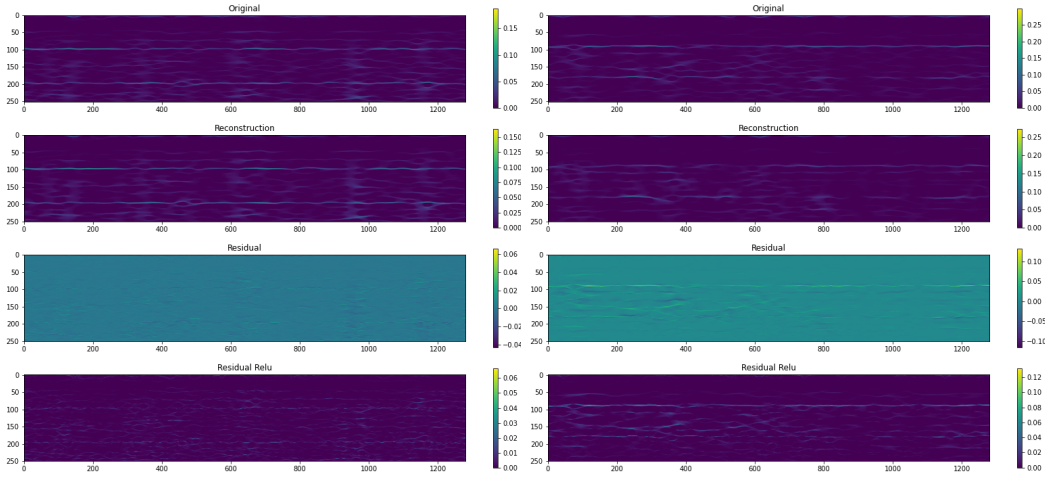
(**a**) (**b**)

Figure 12. Results of the CAE with a spatial latent space: (**a**) train and validation loss histories; (**b**) boxplots representing the distributions of the reconstruction losses (MAE and ReLU MAE) for each round.

The trained model was subsequently tested on the other rounds. Figure 12b shows the boxplots of the MAE and ReLU MAE obtained by the model in all the rounds. According to this model, the "real" rounds are all nominal, whereas both anomaly metrics increase substantially for the "artificial" rounds. Figure 13a,b show the results for the lowest MAE (round 3) and the highest one (modified round 2), respectively. From the comparison of the two ReLU residuals, it can be seen that an almost perfect reconstruction is obtained for round 3, whereas the main frequency component present in the modified round 2 spectrogram is perfectly preserved, meaning that is an anomalous pattern according this CAE.

4.3. VAE with Dense Latent Space

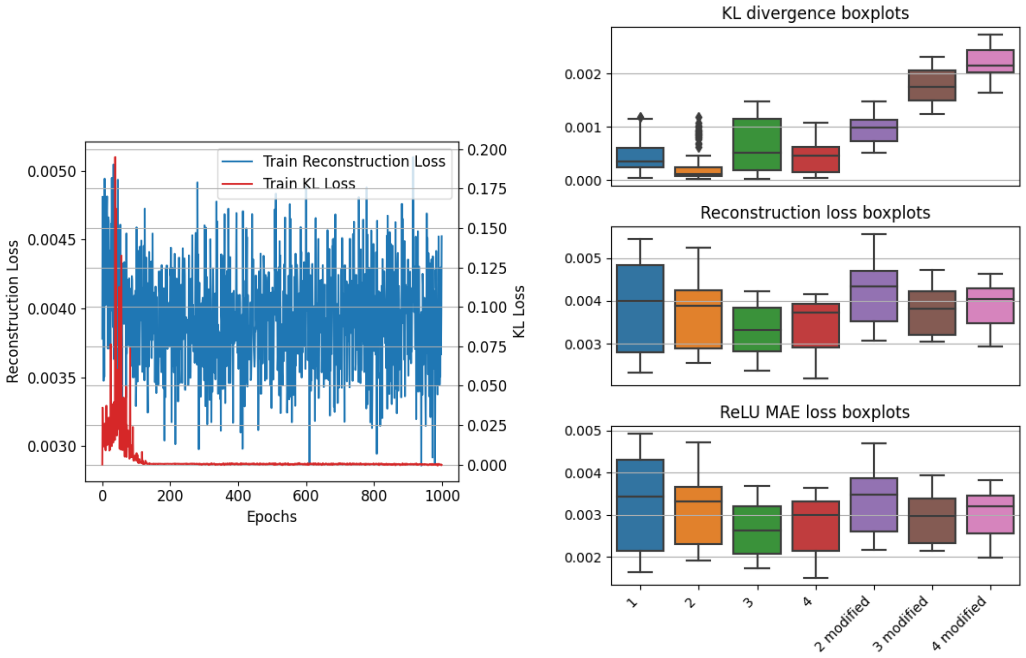
The training time for this model took approximately 5 s per epoch. After evaluating both cycling and monotonic annealing with different combinations of cycle length and max β value, it was noted that the monotonic strategy with a max β of 10^3 yielded a smooth reduction in the overall loss. Figure 14a shows the history of reconstruction and KL loss during training. The boxplots with the KL divergence, MAE, and ReLU MAE obtained when testing the model on all the rounds are represented in Figure 14b. In VAEs, the KL divergence loss can be used as an anomaly score; it appears that the distribution estimated for the "synthetic" rounds progressively deviates from a standard Gaussian distribution.



(**a**) (**b**)

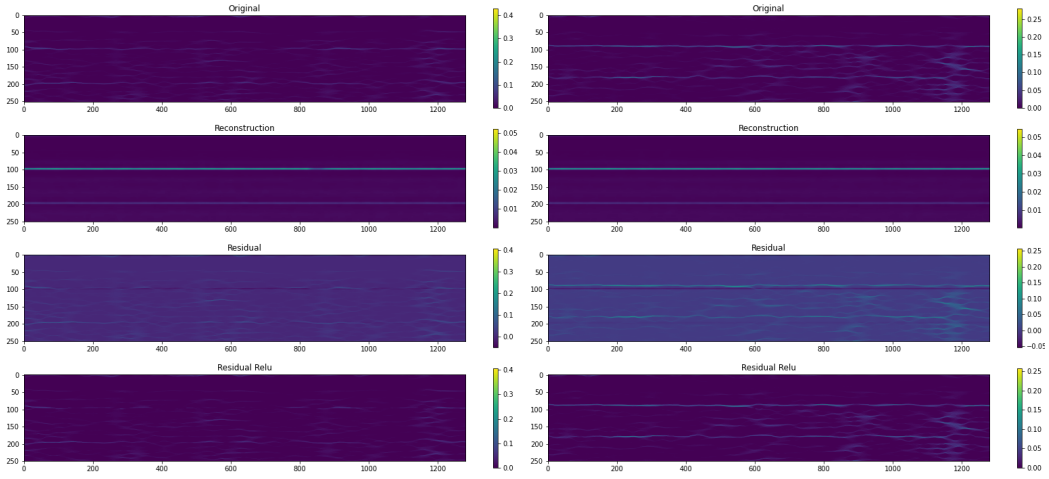
Figure 13. Visualization of the results for the CAE with a spatial latent space, for a random spectrogram extracted from round 3 (**a**) and from round 2 modified (**b**). From top to bottom: original spectrogram, spectrogram reconstructed by the CAE, residual spectrogram, and ReLU residual.

The lowest median for the ReLU MAE is achieved for round 3, whereas the highest is for the modified round 2. One spectrogram for each of these rounds is represented in Figure 15a,b. Two results are quite visible here: the two reconstructed spectrograms are practically the same, and they seem to represent an "average" round 1 spectrogram. This result is noted for more spectrograms taken randomly from any round, which means that this "average" spectrogram is produced by this VAE regardless of the input tensor. The ReLU residual from round 3 (Figure 15a) could be interpreted as having no anomaly identified by the model, whereas the ReLU residual for round 2 modified shows that the original spectrogram is entirely preserved, meaning that it is likely to present anomalous patterns.



(**a**) (**b**)

Figure 14. Results of the VAE with a dense latente space: (**a**) reconstruction and KL loss histories in training; (**b**) boxplots representing the distributions of the reconstruction losses (KL divergence, MAE, and ReLU MAE) for each round.

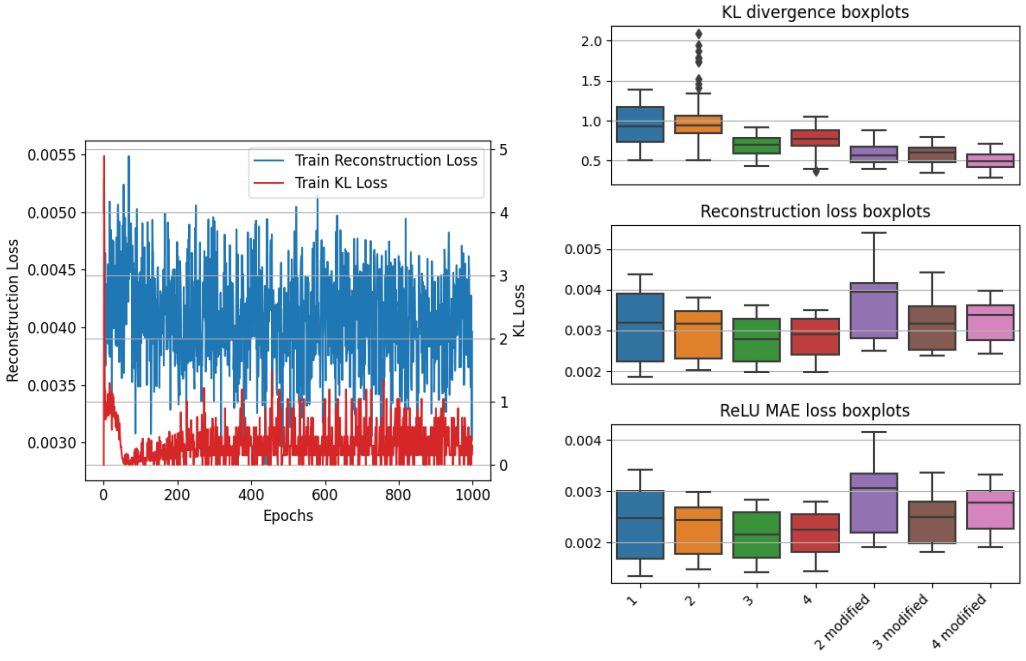


(**a**) (**b**)

Figure 15. Visualization of the results for the VAE with a dense latent space for a random spectrogram extracted from round 3 (**a**) and from round 2 modified (**b**). From top to bottom: original spectrogram, spectrogram reconstructed by the VAE, residual spectrogram, and ReLU residual.

4.4. VAE with Spatial Latent Space

Finally, the VAE with a spatial latent space was trained on round 1, with monotonic annealing and a max β of 10^6, following similar considerations to the ones made for the dense VAE. Also for this model, the training time took 5 s per epoch. The training history is reported in Figure 16a. The KL loss value appears unstable during the training process.



(**a**) (**b**)

Figure 16. Results of the VAE with a spatial latente space: (**a**) reconstruction and KL loss histories in training; (**b**) boxplots representing the distributions of the reconstruction losses (KL divergence, MAE, and ReLU MAE) for each round.

Figure 16b shows the boxplots for the three losses (KL divergence, MAE, and ReLU MAE). In this case, the KL divergence loss decreases with the rounds, whereas the "synthetic" rounds achieve higher reconstruction losses compared to the real rounds, although

the difference is not sharp. Figures 17 and 18 represent the spectrograms for which the lowest and highest median reconstruction loss is achieved by the spatial VAE model, respectively, for round 3 and round 2 modified. The residual spectrograms differ substantially between the two rounds: most of the original spectrogram from round 3 is not present in the ReLU residual, whereas the majority of the spectrogram from round 2 modified is preserved in the ReLU residual. This indicates that the patterns in the spectrogram from round 2 modified are considered anomalous by this model.

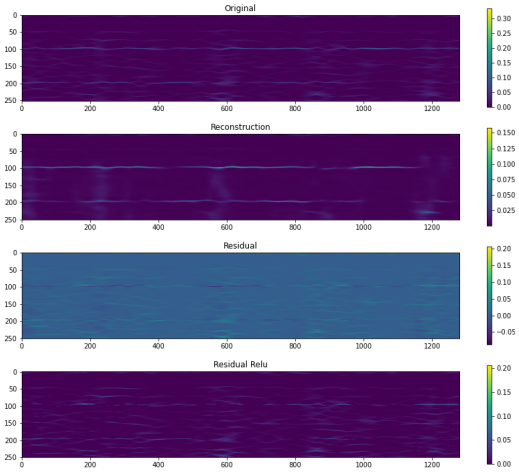


Figure 17. Original spectrogram from round 4, spectrogram reconstructed by the VAE with a spatial latent space, residual spectrogram, and ReLU residual.

Although it appears that the model is able to identify anomalous patterns at the pixel level, this capacity does not reflect on the metrics at the image level (Figure 16b), where none of the metrics substantially increase with the round number.

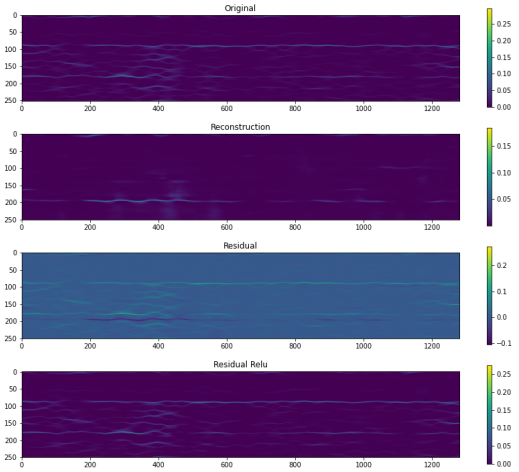


Figure 18. Original spectrogram from round 2 modified, spectrogram reconstructed by the VAE with a spatial latent space, residual spectrogram, and ReLU residual.

4.5. Discussion

By examining the reconstructed spectrograms, it is evident that utilizing a spatial latent space enables accurate image reconstruction. Particularly in the VAE, the dense latent space (Figure 15a,b) proves inadequate for characterizing the spectrograms, as all output tensors exhibit a similar appearance, resembling an "average" and simplified representation of the round 1 spectrograms. This is also reflected in the KL loss achieved for the VAE with a dense latent space, as depicted in Figure 14b. It shows a very low KL divergence only for round 2, with limited median values for rounds 1, 3, and 4, that substantially

increase for all the synthetic rounds. This indicates that the produced distributions become increasingly different from the nominal one with each round. Generally speaking, the poor results of the two VAEs compared to the two CAEs in AD capabilities can be attributed to the fact that reconstruction metrics are not suitable for use with VAEs. Due to the statistical arrangement of their latent space, metrics built in the latent space (such as the KL divergence) or more refined methods based on the latent space distribution should be used instead. This outcome could also be due to a suboptimal choice of the latent space or the training approach. Optimization of both factors should be addressed in future work, as it is evident that, compared to their CAE counterparts, VAEs require dedicated and more refined hyperparameter tuning.

It is also noteworthy that the spatial latent space requires significantly fewer weights—and consequently, less model memory usage—compared to the dense latent space. In fact, better reconstruction losses are achieved with spatial latent spaces within a comparable training time.

Regarding AD at the pixel level, employing the ReLU residual instead of the simple residual tensor results in a more comprehensible representation of the anomalous frequency components in the spectrogram. Additionally, the use of ReLU MAE and simple MAE did not yield significant differences at the image level, regardless of the model considered, as clearly seen in Figures 10b, 12b, 14b and 16b. It is also evident that when relying solely on image-level losses for making informed decisions, the CAE with a spatial latent space is the only model capable of identifying a clear distinction between the "real" and "synthetic" data. Moreover, due to the extensive use of VAEs in the literature for AD in images, it is clear that further work on the metrics and models is required to fine-tune VAEs to optimal working conditions.

Finally, it is important to discuss the practical implementation of this methodology in a real-world industrial scenario. For instance, the duration required to collect a training set comprising signals representative of "healthy" machinery should be assessed beforehand, necessitating a minimum understanding of the degradation velocity. Additionally, if one relies solely on metrics extracted at the image level (such as reconstruction loss or ReLU MAE), a subsequent method should be introduced to define a threshold (or multiple thresholds) beyond which the machinery is considered non-operable. To establish these thresholds, it might be essential to collect more vibrational datasets from similar systems undergoing degradation, thereby compiling a comprehensive collection of data upon which to base the selection of the thresholds. Furthermore, factors such as operational environments and different machinery types might have unpredictable effects on the acquired vibrations and the consequent model behavior. This necessitates the consideration of various possible vibrational datasets in the methodology evaluation, which will enable more robust and generalizable anomaly detection capabilities.

5. Conclusions

In this paper, we have presented a methodology for monitoring machinery based on vibration analysis. The methodology relies on the FSST, followed by the use of a CAE. This paper explores the rationale behind introducing such a methodology: in pixel-level AD of images, unsupervised methods have seen outstanding development over the last decade due to the implementation of convolutional neural networks. Moreover, the research community is gradually acknowledging that unsupervised methodologies should be preferred for monitoring problems in industrial contexts. The use of the SST allows for the production of sparse and clear spectrograms, while the use of AEs and VAEs is explored for both dense and spatial latent spaces. Overall, the superiority of the spatial latent space compared to the dense one is observed in terms of reconstruction accuracy and

AD capabilities, with the CAE with a spatial latent space able to identify as anomalous the synthetic rounds, where the main frequency components have been substantially shifted. Additionally, the VAEs did not yield satisfactory results, likely due to a poor choice of the metrics and non-optimal training process. Finally, it is observed that the use of ReLU residuals could aid the human eye in identifying anomalies present in the spectrogram, although this metric might not be suitable for VAEs.

Further developments are necessary to fine-tune the VAEs and bring this type of generative neural network to optimal conditions for AD purposes. This objective will involve not only a revision of the model's structure, but also of the metrics employed and, likely, part of the methodology. Furthermore, future work should include comparing additional state-of-the-art models in the field of anomaly detection at the pixel level, beyond the four models presented in this paper. This will enable a more comprehensive and favorable comparison.

Author Contributions: Conceptualization, L.R. and F.M.B.; methodology, L.R. and F.M.B.; software, L.R. and F.M.B.; validation, L.R.; formal analysis, L.R.; investigation, L.R. and F.M.B.; resources, S.C.; data curation, L.R.; writing—original draft preparation, L.R.; writing—review and editing, L.R., F.M.B. and S.C.; visualization, L.R.; supervision, S.C.; project administration, S.C.; funding acquisition, S.C. All authors have read and agreed to the published version of the manuscript.

Funding: The authors acknowledge funding from FSE REACT-EU, Programma Operativo Nazionale Ricerca e Innovazione 2014–2020 (CCI2014IT16M2OP005), Azione IV.5 "Dottorati su tematiche Green" (DOT1316301-5).

Data Availability Statement: The original data presented in this study are openly available in a Google Drive repository at https://bit.ly/4ahDnzY (accessed on 26 January 2025).

Acknowledgments: The authors are grateful to the FlexLink company for providing the test rig and continuous support.

Conflicts 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.

Abbreviations

The following abbreviations are used in this manuscript:

AD	Anomaly Detection
AE	Autoencoder
VAE	Variational Autoencoder
CAE	Convolutional Autoencoder
KL	Kullback–Leibler
STFT	Short-Time Fourier Transform
SST	Synchrosqueezing Transform
DWT	Discrete Wavelet Transform
FSST	Synchrosqueezing of a Short-Time Fourier Transform
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
ELBO	Evidence Lower Bound
MAE	Mean Absolute Error
ReLU	Rectified Linear Unit

References

1. Molęda, M.; Małysiak-Mrozek, B.; Ding, W.; Sunderam, V.; Mrozek, D. From Corrective to Predictive Maintenance—A Review of Maintenance Approaches for the Power Industry. *Sensors* **2023**, *23*, 5970. [CrossRef] [PubMed]
2. Yang, S. A condition-based failure-prediction and processing-scheme for preventive maintenance. *IEEE Trans. Reliab.* **2003**, *52*, 373–383. [CrossRef]
3. Rytter, A. Vibrational Based Inspection of Civil Engineering Structures. Ph.D. Thesis, University of Aalborg, Aalborg, Denmark, 1993; p. 206.
4. Worden, K.; Dulieu-Barton, J.M. An overview of intelligent fault detection in systems and structures. *Struct. Health Monit.* **2004**, *3*, 85–98. [CrossRef]
5. Bruwer, M.J.; MacGregor, J.F.; Bourg, W.M. Soft sensor for snack food textural properties using on-line vibrational measurements. *Ind. Eng. Chem. Res.* **2007**, *46*, 864–870. [CrossRef]
6. Bono, F.M.; Radicioni, L.; Cinquemani, S. A novel approach for quality control of automated production lines working under highly inconsistent conditions. *Eng. Appl. Artif. Intell.* **2023**, *122*, 106149. [CrossRef]
7. Randall, R.B. *Vibration-Based Condition Monitoring: Industrial, Automotive and Aerospace Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2021.
8. Matania, O.; Bachar, L.; Bechhoefer, E.; Bortman, J. Signal Processing for the Condition-Based Maintenance of Rotating Machines via Vibration Analysis: A Tutorial. *Sensors* **2024**, *24*, 454. [CrossRef]
9. Stelter, P. Nonlinear vibrations of structures induced by dry friction. *Nonlinear Dyn.* **1992**, *3*, 329–345. [CrossRef]
10. Brunton, S.L.; Kutz, J.N. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*; Cambridge University Press: Cambridge, UK, 2022.
11. Ge, Z.; Song, Z.; Ding, S.X.; Huang, B. Data mining and analytics in the process industry: The role of machine learning. *IEEE Access* **2017**, *5*, 20590–20616. [CrossRef]
12. Eltouny, K.; Gomaa, M.; Liang, X. Unsupervised learning methods for data-driven vibration-based structural health monitoring: A review. *Sensors* **2023**, *23*, 3290. [CrossRef]
13. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef]
14. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 618–626.
15. Qian, J.; Song, Z.; Yao, Y.; Zhu, Z.; Zhang, X. A review on autoencoder based representation learning for fault detection and diagnosis in industrial processes. *Chemom. Intell. Lab. Syst.* **2022**, *231*, 104711. [CrossRef]
16. Baur, C.; Wiestler, B.; Albarqouni, S.; Navab, N. Deep autoencoding models for unsupervised anomaly segmentation in brain MR images. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part I*; Springer: Cham, Switzerland, 2019; pp. 161–169.
17. Bergmann, P.; Fauser, M.; Sattlegger, D.; Steger, C. MVTec AD—A comprehensive real-world dataset for unsupervised anomaly detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 9592–9600.
18. Venkataramanan, S.; Peng, K.C.; Singh, R.V.; Mahalanobis, A. Attention guided anomaly localization in images. In *Computer Vision—ECCV 2020, 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII*; Springer: Cham, Switzerland, 2020; pp. 485–503.
19. Gangloff, H.; Pham, M.T.; Courtrai, L.; Lefèvre, S. Unsupervised Anomaly Detection Using Variational Autoencoder with Gaussian Random Field Prior. In Proceedings of the 2023 IEEE International Conference on Image Processing (ICIP), Kuala Lumpur, Malaysia, 8–11 October 2023; pp. 1620–1624.
20. Nguyen, H.H.; Nguyen, C.N.; Dao, X.T.; Duong, Q.T.; Kim, D.P.T.; Pham, M.T. Variational Autoencoder for Anomaly Detection: A Comparative Study. *arXiv* **2024**, arXiv:2408.13561.
21. Hu, X.; Zhang, X.; Wang, Z.; Chen, Y.; Xia, J.; Du, Y.; Li, Y. Railway switch machine fault diagnosis considering sensor abnormality scenarios. In Proceedings of the 2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), Bilbao, Spain, 24–28 September 2023; pp. 4834–4839.
22. Bono, F.M.; Radicioni, L.; Cinquemani, S.; Bombaci, G. A Comparison of Deep Learning Algorithms for Anomaly Detection in Discrete Mechanical Systems. *Appl. Sci.* **2023**, *13*, 5683. [CrossRef]
23. Wang, P.; Gao, J.; Wang, Z. Time-frequency analysis of seismic data using synchrosqueezing transform. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 2042–2044. [CrossRef]

24. Hazra, B.; Pantula, S.; Narasimhan, S. Novelty detection in airport baggage conveyor gear-motors using Synchro-squeezing transform and Self-organizing maps. In Proceedings of the Annual Conference of the PHM Society, New Orleans, LA, USA, 14–17 October 2013; Volume 5.
25. Hazra, B.; Narasimhan, S. Gearbox fault detection using synchro-squeezing transform. *Procedia Eng.* **2016**, *144*, 187–194. [CrossRef]
26. Hazra, B.; Sadhu, A.; Narasimhan, S. Fault detection of gearboxes using synchro-squeezing transform. *J. Vib. Control* **2017**, *23*, 3108–3127. [CrossRef]
27. Mao, X. A concentrated time-frequency method for reservoir detection using adaptive synchrosqueezing transform. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [CrossRef]
28. Tian, B.; Wen, S.; Li, X.; Ju, J.; Tang, J.; Xiong, N. PFMD: A power frequency magnetic anomaly signal detection scheme based on synchrosqueezed wavelet transform. *Appl. Sci.* **2022**, *12*, 10735. [CrossRef]
29. Rasul, M.; Kawashima, M.; Hoang, K.T. Development of a Deep Learning-Based Anomaly Detection System for Structures. In Proceedings of the International Symposium of the International Federation for Structural Concrete, Istanbul, Turkey, 5–7 June 2023; pp. 1578–1588.
30. Ruff, L.; Kauffmann, J.R.; Vandermeulen, R.A.; Montavon, G.; Samek, W.; Kloft, M.; Dietterich, T.G.; Müller, K.R. A unifying review of deep and shallow anomaly detection. *Proc. IEEE* **2021**, *109*, 756–795. [CrossRef]
31. Hatami, N.; Gavet, Y.; Debayle, J. Classification of time-series images using deep convolutional neural networks. In Proceedings of the Tenth International Conference on Machine Vision (ICMV 2017), Vienna, Austria, 13–15 November 2017; Volume 10696, pp. 242–249.
32. Yang, C.L.; Yang, C.Y.; Chen, Z.X.; Lo, N.W. Multivariate time series data transformation for convolutional neural network. In Proceedings of the 2019 IEEE/SICE International Symposium on System Integration (SII), Paris, France, 14–16 January 2019; pp. 188–192.
33. Daubechies, I.; Maes, S. A nonlinear squeezing of the continuous wavelet transform based on auditory nerve models. In *Wavelets in Medicine and Biology*; Routledge: London, UK, 2017; pp. 527–546.
34. Behera, R.; Meignen, S.; Oberlin, T. Theoretical analysis of the second-order synchrosqueezing transform. *Appl. Comput. Harmon. Anal.* **2018**, *45*, 379–404. [CrossRef]
35. Radicioni, L.; Bono, F.M.; Cinquemani, S. Application of synchrosqueezing transform and autoencoders for monitoring of production systems. A case study on plastic chain conveyor systems. In Proceedings of the 2024 20th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA), Genova, Italy, 2–4 September 2024; pp. 1–7.
36. Asperti, A.; Trentin, M. Balancing reconstruction error and kullback-leibler divergence in variational autoencoders. *IEEE Access* **2020**, *8*, 199440–199448. [CrossRef]
37. Kingma, D.P. Auto-encoding variational bayes. *arXiv* **2013**, arXiv:1312.6114.
38. Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.P.; Glorot, X.; Botvinick, M.M.; Mohamed, S.; Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. In Proceedings of the 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, 24–26 April 2017.
39. Fu, H.; Li, C.; Liu, X.; Gao, J.; Celikyilmaz, A.; Carin, L. Cyclical annealing schedule: A simple approach to mitigating kl vanishing. *arXiv* **2019**, arXiv:1903.10145.
40. Bowman, S.R.; Vilnis, L.; Vinyals, O.; Dai, A.M.; Jozefowicz, R.; Bengio, S. Generating sentences from a continuous space. *arXiv* **2015**, arXiv:1511.06349.
41. Radicioni, L.; Bono, F.M.; Cinquemani, S. On the use of vibrations and temperatures for the monitoring of plastic chain conveyor systems. *Mech. Syst. Signal Process.* **2025**, *223*, 111935. [CrossRef]
42. Mordvintsev, A.; Abid, K. Opencv-Python Tutorials Documentation. 2014. Available online: https://docs.opencv.org/3.4/da/d54/group__imgproc__transform.html (accessed on 9 February 2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.