A Supervised Approach for Corrective Maintenance Using Spectral Features from Industrial Sounds

Luana Gantert[1], Matteo Sammarco[2], Marcin Detyniecki[2], and Miguel Elias M. Campista[1]
[1]*GTA-PEE/COPPE-DEL/Poli – Universidade Federal do Rio de Janeiro (UFRJ)* – Rio de Janeiro, Brazil
[2]*AXA* – Paris, France
{gantert, miguel}@gta.ufrj.br, {matteo.sammarco, marcin.detyniecki}@axa.com

***Abstract*—The fourth industrial revolution makes extensive use of IoT, AI, and smart sensors for improved automation, safety, production, and prognostics, and health management. In this paper, we address corrective maintenance based on fault recognition relying on sounds produced by machine components. Different spectral features are extracted from industrial sounds and are used as input of supervised learning algorithms for classification between normal and abnormal operations. Experiments using the MIMII (Malfunctioning Industrial Machine Investigation and Inspection) dataset, which contains sound samples produced by pump, slide rail, valve, and fan components, reveals promising results based on the f1-score. We also evaluate the impact of the different spectral features considered, confirming their incremental impact. Finally, we compare our proposal with a baseline alternative from the literature, which employs unsupervised learning and Mel-spectrogram conversion. Our approach improves the AUC (Area Under the Curve) metric by up to 39.5% compared with the baseline approach.**

***Index Terms*—IoT, machine learning, maintenance engineering**

I. Introduction

Prognostics and Health Management (PHM) is emerging in the industry as an intelligent process for component health monitoring. In PHM, detection and interpretation of different parameters play a key role in fault detection and condition monitoring, diagnosis, and prognostics in failure modes [1]. All these tasks are important for either corrective or preventive maintenance. The former only happens when a component failure is spotted and aims to maximize components lifespan, whereas the latter is based on scheduled operations aiming to anticipate failure occurrences. These approaches have pros and cons on costs and reliability [2], making search for the optimal point between them a target for PHM [1].

Detecting fault conditions in machines or system components is fundamental to generate alarms and determine the state of PHM systems. This can be handled by classifiers, which typically follow a well-known workflow consisting of data acquisition and analysis. In Industry 4.0, this workflow can be considered a trend as machine-learning-based approaches have been extensively deployed in smart manufacturing systems [3], [4]. A recent approach based on image processing is becoming more and more popular in corrective maintenance. Classification tasks based on Mel-spectrograms built from industrial sounds can benefit from well-known classifiers, such as Convolutional Neural Networks (CNN). The downside, however, is on processing requirements for training and inference as the neural networks used may become quickly complex. For instance, ResNet50 is a Deep Residual Network for image recognition used in the DCASE Challenge 2020 by Giri et al [5] consisting of 23 million trainable parameters. Such an amount of parameters needs an adequate size of training data to avoid overfitting and, as a consequence, a low generalization performance. Especially in the context of PHM relying on audio signals for industrial machinery corrective maintenance, abnormal sounds are difficult to catch and record as they cannot be artificially reproduced. On the other hand, normal operating sounds are always similar by design, which does not provide a good variety of input and intensify the risk of overfitting. In addition to the computational and training issues, the data produced during normal behavior can be orders of magnitude larger than the data produced during fault conditions. This imbalanced nature imposes an additional challenge that must be tackled by classifiers.

In this work, for all the reasons exposed above, we propose the use of spectral features instead of Mel-spectrograms to recognize fault events for corrective maintenance in industrial scenarios. Thus, it is possible to reduce the complexity of machine-learning-based solutions and, as consequence, the delay to activate alarms based on known fault conditions. We use as the input of different supervised algorithms, instead of images, tuples containing spectral features from different industrial sounds.

We conduct experiments using the MIMII (Malfunctioning Industrial Machine Investigation and Inspection) dataset, which contains sound samples produced from the operation of different machine components [6]. The MIMII dataset contains the sounds of four different machine components: pump, slide rail, valve, and fan. Each component has samples recorded from four different machines, with three different levels of signal-to-noise (SNR) ratios to simulate real industrial environments. This dataset is unbalanced, with far fewer samples of fault conditions. We circumvent this issue by oversampling the anomalous samples.

Our analysis starts by using the f1-score to evaluate the performance of four different supervised algorithms: Support

This work was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. It was also supported by CNPq, FAPERJ Grants E-26/211.144/2019 and E-26/202.689/2018, and FAPESP Grant 15/24494-8.

Vector Machine (SVM), Multilayer Perceptron (MLP), Random Forrest (RF), and Naive Bayes (NB), chosen given their popularity in classification tasks. Besides, these algorithms do not require a very large amount of data, suiting well our case as the dataset used has about 30 000 samples for each SNR for all components. Then, we evaluate the importance of the spectral features with the two algorithms that have shown the best performance in the previous analysis. For this evaluation, we use the AUC (Area Under the Curve) metric, which indeed confirms that all selected spectral features contribute to the system performance.

In the end, we compare our proposal with the results obtained by the authors of the MIMII dataset, who evaluated the performance of a Dense Autoenconder Network, using Mel-spectrograms as input. For this particular problem, our proposed supervised approach shows a better performance according to the AUC metric, independently of the scenario and component to classify. For example, in the best case, we improve the AUC compared with the baseline approach of 39.5%, from 0.665 to 0.928, for the pump class with SNR $= -6\,\text{dB}$.

This paper is organized as follows. Section II reviews the works focusing on machine-learning approaches for the detection of events or failures.Section III presents our proposal while Section IV describes more in detail the dataset and the workflow implementation. Section V provides the obtained results. Finally, in Section VI, we draw conclusions as well as future plans.

II. Related Work

Machine-learning approaches have been broadly applied to industrial scenarios. Just to cite one example, Zhang et al. provide a Deep Belief Network model for monitoring the health status information of rolling bearings from vibration data [3]. Many other possibilities are surveyed by Diez-Olivan et al. who compare previous maintenance works based on artificial neural networks and k-nearest neighbors in different industrial sectors [4]. We focus, however, on works addressing corrective maintenance based on industrial sounds.

Purohit et al. are the authors of the MIMII dataset containing normal and anomalous samples of industrial sounds and, consequently, they provide the first analysis using a Dense Autoencoder Network model [6]. Their network model has five layers, with 8 neurons in the central layer and 64 neurons in all the other layers, and used Mel-spectrograms as input. We consider this work as the baseline of our performance evaluation in Section V.

The work from Purohit et al. inspired extensions on the DCASE Challenge 2020, which also proposed unsupervised approaches for sound classification. In that event, the best performance on general audio event recognition tasks also used Mel-spectrogram images, but as input to MobileNetV2 and ResNet50 models [5]. Daniluk et. al. were among the top-ranked in the same challenge with a network based on Variational and Conditioned Auto-Encoders. On top of that, a denoise network based on Deep Complex U-Net is used in the preprocessing step to reduce background noise for the different SNR [7]. Primus comes third in the same ranking [8]. His proposition is based on a Residual Network (ResNet). Even though these works could achieve promising results, they all use DNN models requiring a Mel-spectrogram image as input and a very large amount of training data to avoid the risk of overfitting, regardless of knowledge transfer, knowledge distillation, or regularization techniques. We rely on a model with much less trainable parameters and features as input extracted from the audio signal and its frequency domain representation. For the same reasons, Sammarco and Detyniecki adopted a similar approach for the detection of dangerous events in the road safety context [9].

III. Corrective Maitenance Using Spectral Features

Figure 1 provides a general picture of the proposed system based on supervised learning. The final goal is to provide corrective maintenance based on fault recognition. To accomplish that we propose, a classification approach that takes machine component sounds as input. We begin extracting spectral features from the original dataset. The output of this initial step is a dataset with the selected metadata. This converted dataset is split into three sets for training, validation, and test. The workflow then reaches a bifurcation where the model is produced using as input the validation and the training sets. The training and the validation sets go through an oversampling step to balance the number of anomalous and normal samples. Next, they are scaled and relabeled using numerical notation to represent abnormal and normal samples. Finally, the supervised learning model is trained and further evaluated using the test dataset. Note that the test dataset has previously gone through the same data transformation we use for the validation and training sets. We detail the main tasks, spectral features extraction, and sound classification, in the next sections.

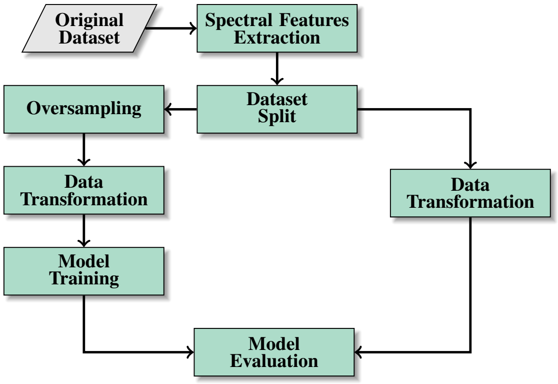


Fig. 1: The system proposed for corrective maintenance using spectral features extracted from sounds as input.

A. Selected Spectral Features

Audio spectral features are extracted by converting time-based audio signals to the frequency domain. Let us consider

$x(t)$ as a discrete audio signal of N samples. It is transposed to the frequency domain through the following process: $x(t)$ is split into fixed M-size smaller frames, with $M << N$. Then, the Discrete Fourier Transform (DFT) is applied to each frame returning an array of coefficients. Let us now consider $X_i(k), k = 1, \dots, M$ as the DFT coefficients magnitude for the i^{th} frame.

Features extracted by the frequency domain are numerous and have been used in audio signal processing for different purposes. A thorough list of spectral features can be found in the work by Chaki [10]. In the current paper, we narrow down the list to features largely adopted in the literature for audio event detection and music genre classification tasks as follows:

- **Chroma**: The chromagram vector represents the entire spectrum by grouping the whole energy into the twelve different musical pitch classes of an equal-tempered scale. This feature can represent in a concise form the harmonic and musical signals. Each element of the vector is the mean of DFT coefficients:

$$v_k = \sum_{f \in F_k} \frac{X_i(f)}{|F_k|}, \quad k \in [1, 12],$$

where F_k is the set of frequencies included in the same group;

- **Mel-frequency cepstral coefficients (MFCC)**: These coefficients can describe phonetically the sound envelope. The envelope is the instantaneous amplitude of an oscillating signal. This feature is widely used in speech recognition for its compact representation of the energy spectrum expressed on a Mel-frequency scale [11].

$$Mel(f) = 2595 \log_{10}\left(1 + \frac{f}{100}\right).$$

MFCCs are computed as the inverse discrete cosine transform (DCT) of cepstrum powers at each Mel frequency:

$$MFCC = \sqrt{\frac{2}{M_{mfcc}}} \sum_{m=1}^{M_{mfcc}} \log(\tilde{X}_m(t)) \cos(\frac{c(n - \frac{1}{2})}{M_{mfcc}}),$$

where M_{mfcc} is the number of Mel frequency bands (usually $M_{mfcc} = 12$), $\tilde{X}_m(t)$ is the signal energy in the m^{th} Mel frequency band and $c \in [1 - -M_{mfcc}]$ is the index of the cepstrum coefficient.

- **Spectral Centroid (SC)**: the magnitude of the spectrum center of gravity. This feature is associated with the subjective brightness idea of sound intensity. Signals with higher frequencies tend to be clearer. For the i^{th} frame, the spectral centroid is the average of frequencies present in the signal, weighted by their amplitudes:

$$SC_i = \frac{\sum_{k=1}^{M} k X_i(k)}{\sum_{k=1}^{M} X_i(k)};$$

- **Spectral Bandwidth (SB)**: Spectral bandwidth is calculated considering the spectral centroid SC as follows:

$$SB = \sqrt{\sum_{k=1}^{M} X_i(k)(f_k - SC)^2};$$

- **Spectral roll-off (SR)**: the frequency below which 85% of the spectral magnitude is; SR is the frequency that satisfies the following relation:

$$\sum_{k=1}^{SR} X_i(k) = 0.85 \sum_{k=1}^{M} X_i(k).$$

- **Zero Crossing Rate (ZCR)**: The ZCR counts the number of times an audio signal waveform crosses the zero, and it is directly computed from the time domain as:

$$ZCR = \frac{1}{2N} \sum_{t=0}^{N-2} |sgn(x(t+1)) - sgn(x(t))|,$$

where $sgn(x(t)) = \begin{cases} 1, & \text{if } x(t) \geq 0, \\ -1, & \text{if } x(t) < 0. \end{cases}$.

B. Abnormal Sound Classification

The dataset is split into three sets: training, validation, and test. We tune hyperparameters for the training and validation sets, and then the model is built. These hyperparameters depend on the supervised approach employed, which will be discussed in the next section. As the dataset is imbalanced, we add anomalous samples in these sets before training the model. We randomly resample the abnormal samples in training and validation sets. The test set is used later for performance evaluation.

In the data transformation step, we use Standard Scaler on the data, i.e., all the sample values are transformed to have a mean equal to 0 and a standard deviation equal to 1. Then, we put the variables into the same scale and enable comparison scores between all the different features. The labels are converted into a binary number, with normal status equal to 0 and abnormal status equal to 1.

We select four popular techniques to build models and examine to which extent each alternative can improve the results obtained in [6], which we consider our baseline. The supervised approaches are Naive Bayes (NB), Random Forest (RF), Support Vector Machine (SVM), and Multilayer Perceptron (MLP).

- **NB**: The Naive Bayes method is a type of probabilistic classifier based on Bayes' theorem. These classifiers use the "naive" assumption of independence between features.
- **RF**: The method consists of constructing decision trees and combining them to improve performance upon choosing the class that appears most often as an output of the individual trees.
- **SVM**: This is a technique for pattern recognition that looks for the best hyperplane to separate different classes.

We implement an SVM model to malfunctioning classification by the performance in audio analysis works [12], [13].

- **MLP**: A class of feedforward artificial neural network. Our proposal is based on an MLP network with three hidden layers.

IV. Dataset and Workflow Implementation

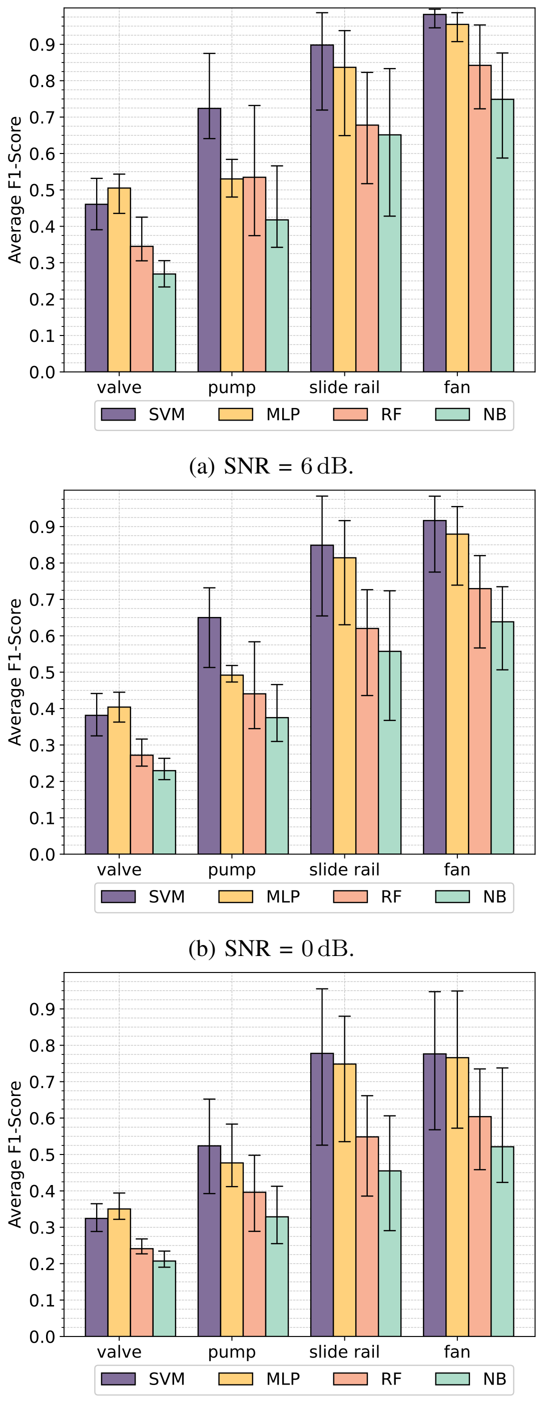
In this section, we describe all the details regarding the dataset used and the workflow implementation for the sake of reproducibility. We used the MIMII (Malfunctioning Industrial Machine Investigation and Inspection) dataset, which contains sound samples from different machines [6]. This dataset contains samples of the normal and anomalous behavior of four types of industrial machines: fan, valve, pump, and slide rails. For each type of machine, there are 4 different datasets. Hence, there is data from 4 different fans, 4 different valves, etc. This dataset is part of the Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge 2020, and the original task consisted of detecting anomalies using unsupervised approaches. The dataset used has also samples from all different components with three different Signal-to-Noise (SNR) ratios, as the correct operation of classifiers may be affected by additional background noise. Noise is recorded in multiple real factories and mixed with the target machine sound. In total, the dataset includes 26 092 normal condition samples and 6065 anomalous ones roughly equally distributed over the 4 categories. The duration of each sound sample is 10 seconds.

TABLE I: MLP hyperparameters.

Hyperparameter	**MLP**
Number of layers	1 input, 3 hidden, 1 output
Nodes per layer	(25, 64, 32, 16, 1)
Activation function	(-, ReLu, ReLu, ReLu, Sigmoid)
Type of layers	Dense
Loss function	Binary Crossentropy
Optimizer	Adam
Batch size	15
Epochs	10

As regarding the workflow depicted in Figure 1, we first load the samples and extract the spectral features using the python Librosa library for audio processing [14]. The next blocks shown in Figure are repeated in 10 independent rounds for each classifier. We then split the dataset into a proportion of 49%, 21%, 40%, regarding training, validation, and test datasets. The oversampling was performed using the homonym function from the Pandas library. Also, the Standard Scaler, the Label Encoder, and all the supervised classifiers are instantiated from the python scikit-learn library classes.

We used the SVM, RF, and NB models from python scikit-learn library. In RF, the only hyperparameter configured was the maximum depth of the tree that we made equal 2. In SVM and NB, we used the default configuration. The MLP was run using the Keras library from python and the hyperparameters are summarized in Table I. Its architecture includes 4289 trainable parameters in total.



(a) SNR = 6 dB.

(b) SNR = 0 dB.

(c) SNR = -6 dB.

Fig. 2: F1-Score comparing four supervised models.

The metrics for performance evaluation are obtained from each respective library (scikit-learn for SVM, RF and NB, and Keras for MLP) and the figures were generated using python library matplotlib.

V. Performance Evaluation

We divide our evaluation into three parts: first, we analyze the performance of the different supervised learning algorithms using the f1-score; then we focus on the top-2 supervised approaches to evaluate the impact of the used spectral features; finally, we compare our proposal to the baseline approach seen in [6] with the same metric used by the authors: the AUC.

The AUC metric is obtained from the Receiver Operating Characteristic (ROC) curve. The ROC curve is plotted using the true positive rate on the y-axis and the false positive rate on the x-axis. The AUC ranges from 0 to 1, being 1 the best performance and 0, otherwise.

It is worth mentioning that all results are obtained from averages computed over individual machines of the same type, for instance, the average over the 4 pumps and the 4 valves, etc. The metric for each individual machine is the average obtained after 10 runs. Also, whenever present, the vertical line indicates the best and the worst result achieved with an individual machine of the same type.

A. Supervised learning performance

We use the f1-score to evaluate the performance of the supervised algorithms because this metric represents both the recall and precision metrics with a single value. Moreover, we separately analyze the impact of background noise on the classification performance for three SNR. Figure 2 shows the f1-score average obtained for each sound class, for the four supervised approaches, and the three SNR. In Figure 2a, the f1-score for the SNR equal to $6\,\mathrm{dB}$ is higher using SVM, except the valve, when MLP is the better choice. The same behavior happens in SNR equal to $0\,\mathrm{dB}$ and $-6\,\mathrm{dB}$, shown in Figure 2b and Figure 2c, respectively.

B. Spectral features impact

We evaluate the impact of the spectral features considering the two best performing algorithms (SVM and MLP) and the AUC metric. This evaluation is presented in Table II for MLP and in Table III for SVM. The analysis is conducted in an incremental mode: starting using only the Chroma feature we add the other features one by one throughout the columns immediately at the right (+SC means Chroma+SC, +SB means Chroma+SC+SB, and so on). All the scenarios show that by putting together all the pre-selected features we achieve the best result: each new feature brings a substantial improvement, independently of the order. In particular, the MFCC feature is especially effective when the background noise is more intense.

C. Comparison results with the baseline approach

We use the AUC metric for comparison with the baseline approach as authors in [6] provide this metric in their paper. Despite the f1-score, the AUC gives also a better clue about the model performance for different precision-recall trade-off values. Thus, for a specific tradeoff, a model can show a better performance than another, while taking into consideration all the values, i.e. the AUC, the result can be different.

TABLE II: Spectral features impact using MLP.

Class	**SNR**	**Chroma**	**+SC**	**+SB**	**+SR**	**+ZCR**	**+MFCCs**
Fan	6 dB	0.798	0.931	0.933	0.948	0.965	**0.997**
	0 dB	0.743	0.835	0.845	0.843	0.868	**0.977**
	−6 dB	0.644	0.743	0.75	0.762	0.77	**0.917**
Pump	6 dB	0.777	0.865	0.913	0.932	0.933	**0.983**
	0 dB	0.708	0.807	0.862	0.879	0.893	**0.966**
	−6 dB	0.61	0.652	0.764	0.778	0.798	**0.928**
Slider	6 dB	0.732	0.836	0.847	0.858	0.897	**0.994**
	0 dB	0.664	0.771	0.79	0.795	0.827	**0.985**
	−6 dB	0.627	0.703	0.731	0.737	0.751	**0.961**
Valve	6 dB	0.519	0.583	0.642	0.656	0.7	**0.929**
	0 dB	0.53	0.543	0.576	0.561	0.597	**0.842**
	−6 dB	0.509	0.546	0.564	0.589	0.595	**0.766**

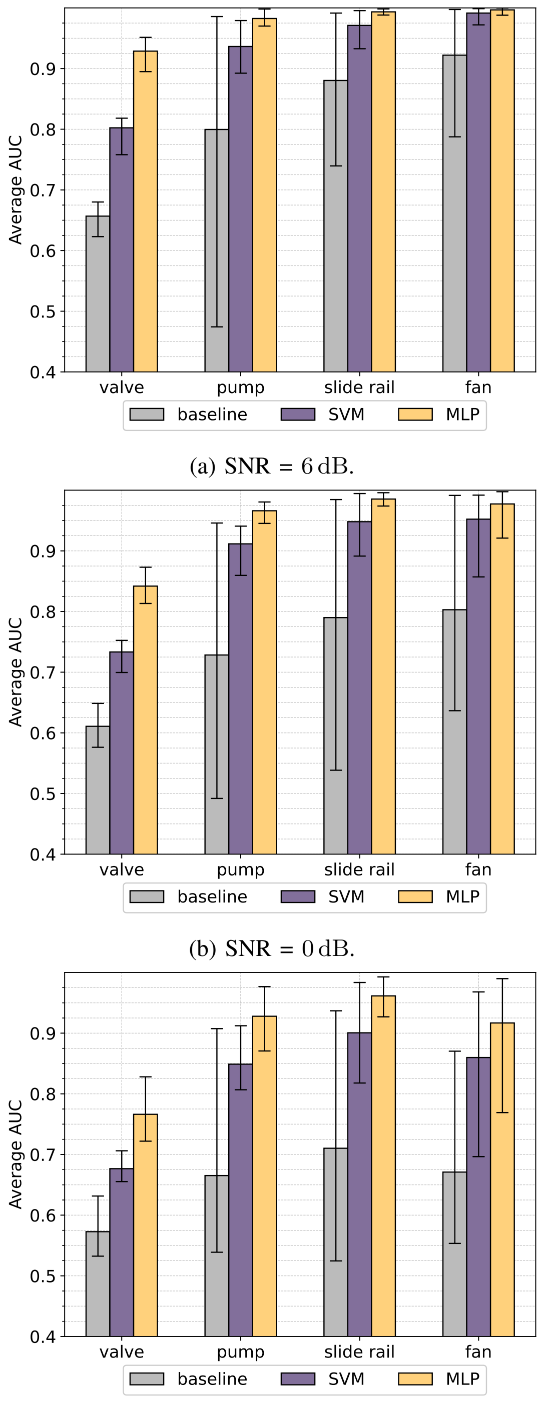
TABLE III: Spectral features impact using SVM.

Class	**SNR**	**Chroma**	**+SC**	**+SB**	**+SR**	**+ZCR**	**+MFCCs**
Fan	6 dB	0.763	0.86	0.862	0.869	0.89	**0.991**
	0 dB	0.717	0.786	0.779	0.789	0.804	**0.952**
	−6 dB	0.628	0.672	0.689	0.691	0.704	**0.86**
Pump	6 dB	0.716	0.778	0.829	0.821	0.814	**0.936**
	0 dB	0.661	0.71	0.776	0.786	0.786	**0.912**
	−6 dB	0.586	0.615	0.682	0.69	0.719	**0.849**
Slider	6 dB	0.7	0.79	0.785	0.792	0.821	**0.971**
	0 dB	0.634	0.719	0.727	0.734	0.755	**0.948**
	−6 dB	0.597	0.663	0.683	0.693	0.705	**0.901**
Valve	6 dB	0.524	0.554	0.603	0.614	0.628	**0.802**
	0 dB	0.512	0.525	0.55	0.542	0.547	**0.733**
	−6 dB	0.509	0.537	0.539	0.547	0.56	**0.677**

We get then the AUC average for each of the three classifiers for all the four types of equipment. The AUC average for the different equipment types is obtained by the AUC average of the results obtained with the individual machines, as explained at the end of Section IV. Figure 3a compares the average AUC for each equipment type when SNR $=$ 6dB. In this scenario, the MLP trained with the spectral features achieves a better performance than the baseline and the SVM model. The minimum and the maximum values for the metric are closer too, indicating less variation in the classification results. The baseline presents the worst performance, presenting the smallest AUC for the valve classification and the highest variation for the pump. The same pattern is found when SNR $=$ 0dB in Figure 3b. At last, Figure 3c shows the SNR $= -$6dB, where the best performance for all the classifiers is in slide rail.

VI. Conclusion

In this work, we address the problem of machine components fault recognition for corrective maintenance through audio analysis. In particular, we focus on the valve, fan, slide rail, and pump as the open MIMII dataset provides normal and anomalous sounds for these four components. While DNN is more and more employed for audio event recognition tasks, they need a large dataset. In the industrial context, a machine component working normally emits the same sound which reduces the variety of negative samples. On the other hand, positive samples are difficult to record or to artificially generate. This makes the use of DNN impracticable without overfitting. For these reasons, we adopt a mixed approach extracting features from the audio signals and feeding an MLP



(a) SNR = 6 dB.

(b) SNR = 0 dB.

(c) SNR = −6 dB.

Fig. 3: AUC comparing our approach with the baseline model.

with a relatively small number of parameters. Classification results (namely SVM) show a better AUC performance concerning classic machine learning models applied to the same dataset while keeping a low number of model parameters for a good generalization.

We plan to evaluate different classification approaches and, in particular, the use of a multi-class recognition able to simultaneously detect faults for different components. Also, we plan to reproduce the same workflow for other different critical scenarios such as those found in vehicular safety.

References

[1] A. J. Guillén, A. Crespo, M. Macchi, and J. Gómez, “On the role of prognostics and health management in advanced maintenance systems,” *Production Planning & Control*, vol. 27, no. 12, pp. 991–1004, 2016.

[2] J. A. Erkoyuncu, S. Khan, A. L. Eiroa, N. Butler, K. Rushton, and S. Brocklebank, “Perspectives on trading cost and availability for corrective maintenance at the equipment type level,” *Reliability Engineering & System Safety*, vol. 168, no. 1, pp. 53–69, 2017.

[3] C. Zhang, Y. Zhang, C. Hu, Z. Liu, L. Cheng, and Y. Zhou, “A novel intelligent fault diagnosis method based on variational mode decomposition and ensemble deep belief network,” *IEEE Access*, vol. 8, no. 1, pp. 36 293–36 312, 2020.

[4] A. Diez-Olivan, J. Del Ser, D. Galar, and B. Sierra, “Data fusion and machine learning for industrial prognosis: Trends and perspectives towards industry 4.0,” *Information Fusion*, vol. 50, no. 1, pp. 92–111, 2019.

[5] R. Giri, S. V. Tenneti, F. Cheng, K. Helwani, U. Isik, and A. Krishnaswamy, “Unsupervised anomalous sound detection using self-supervised classification and group masked autoencoder for density estimation,” Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE 2020 Challenge), Tech. Rep., 2020.

[6] H. Purohit, R. Tanabe, T. Ichige, T. Endo, Y. Nikaido, K. Suefusa, and Y. Kawaguchi, “MIMII Dataset: Sound dataset for malfunctioning industrial machine investigation and inspection,” in *4th Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE 2019)*, Oct. 2019, pp. 209–213.

[7] P. Daniluk, M. Goździewski, S. Kapka, and M. Kośmider, “Ensemble of auto-encoder based and wavenet like systems for unsupervised anomaly detection,” Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE 2020 Challenge), Tech. Rep., 2020.

[8] P. Primus, “Reframing unsupervised machine condition monitoring as a supervised classification task with outlier-exposed classifiers,” Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE 2020 Challenge), Tech. Rep., 2020.

[9] M. Sammarco and M. Detyniecki, “Crashzam: Sound-based car crash detection,” in *Proceedings of the 4th International Conference on Vehicle Technology and Intelligent Transport Systems, VEHITS 2018, Funchal, Madeira, Portugal, March 16-18, 2018*, M. Helfert and O. Gusikhin, Eds. SciTePress, 2018, pp. 27–35. [Online]. Available: https://doi.org/10.5220/0006629200270035

[10] J. Chaki, “Pattern analysis based acoustic signal processing: a survey of the state-of-art,” *International Journal of Speech Technology*, pp. 1–43, 2020.

[11] F. Zheng, G. Zhang, and Z. Song, “Comparison of different implementations of MFCC,” *Journal of Computer Science and Technology*, vol. 16, no. 6, pp. 582–589, 2001.

[12] M. Z. Anwar, Z. Kaleem, and A. Jamalipour, “Machine learning inspired sound-based amateur drone detection for public safety applications,” *IEEE Communications Surveys & Tutorials*, vol. 68, no. 3, pp. 2526–2534, 2019.

[13] İlke Kurt, S. Ulukaya, and O. Erdem, “Musical feature based classification of parkinson’s disease using dysphonic speech,” in *41st International Conference on Telecommunications and Signal Processing (TSP)*, Jul. 2018, pp. 1–4.

[14] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenbergk, and O. Nieto, “librosa: Audio and music signal analysis in python,” in *14th Python in Science Conference (SCIPY 2015)*, Jul. 2015, pp. 18–24.