Pump Fault Classification based on Autoencoding Convolutional Neural Network Residuum

Iulian Vasiliev
Department of Automation and Electrical Engineering
Dunărea de Jos University of Galati
Galati, Romania
Redans SRL, Galati, Romania
iulian.vasiliev@ugal.ro

Laurentiu Frangu
Department of Electronics and Telecommunications
Dunărea de Jos University of Galati
Galati, Romania
Redans SRL, Galati, Romania
laurentiu.frangu@ugal.ro

Mihai Lucian Cristea
Redans SRL, Galati, Romania
mihai@redans.eu

Mihai Cristian Costea
Redans SRL, Galati, Romania
mihaic@redans.eu

***Abstract*—This paper deals with the fault classification of centrifugal pumps, based on the residuum between the output and the input of an Autoencoding Convolutional Neural Network previously trained for abnormal behaviour detection. The proposed classification method performs a dimensional reduction of the residuum vector using Principal Component Analysis, and then, based on the first 3 principal components, classifies the data, using a simple rule-based algorithm, in one of the classes: normal, clogged filter, broken fan blade, detached rotor section and other fault source. The classification method proved to be reliable in an industrial application, providing a 90% correct identification of the machine condition.**

Keywords—fault classification, predictive diagnosis, machine learning, convolutional neural network, vibration analysis, principal component analysis

I. Introduction

The Fourth Industrial Revolution, a concept introduced by Klaus Schwab in 2016 [1], is the trend of 21st century industry towards the usage of advanced computing systems, such as big data, IoT, IIoT, cloud computing, cognitive computing, or artificial intelligence, in the manufacturing process. One of usages of such technologies is the predictive maintenance, the identification of machinery issues in real time, reducing the operational costs by (1) avoiding down-time of the plant and (2) using planned maintenance at low-cost because the operators know beforehand when the machinery will fail. A large part of the equipment used in industry is represented by rotating machinery, and, in consequence, predictive maintenance of such equipment plays a vital role in the manufacturing costs, making the subject of fault detection and classification of rotating machinery an important one through companies and research groups. It must be noted that, because rotating machineries have a large diversity of usages and operating conditions, monitoring of such equipment does not rely on a universal method, but rather specific methods for the application should be implemented. In consequence, this paper deals with fault classification of centrifugal pumps, driven by asynchronous motors. The centrifugal pumps are monitored by accelerometers which provide vibration data. Such data is relevant to the status of the machinery and have been used previously for fault detection using Autoencoding Convolutional Neural Network [2].

The scientific literature treating this subject is very rich, presenting methods of various complexities. Many papers present only fault detection methods to detect abnormal behaviour of the machinery, without treating the fault classification. For example, in [3], the authors present a general view of change detection using vibration analysis, introducing a set of specialized functions for behavioural change detection and segmentation that are using classical analysis in both the time and frequency domain. Other methods used for fault detection include, but are not limited to, continuous wavelet transform [4], bispectral analysis [5], but also algorithms from the machine learning family, ranging from simple feedforward Neural Networks (NN) [6], [7] to more complex solutions involving Convolutional Neural Networks (CNN) [8], [9]. Recent approaches of abnormal behaviour or fault detection propose the usage of Autoencoding NN for diagnosis [10], [11], [2]. Such approaches require only a set of normal behaviour data for training, the NN being able to learn the signature of the normal functioning. When faced with abnormal data, the Autoencoding NN should be able to encode and internally represent the pump signature, removing the components added by the fault. This way, the abnormal functioning can be detected from the residuum between the output and the input data.

Many other recent papers are presenting solutions for fault classification of rotating machinery using a various range of solutions, most of them based on vibration data. In one of the solutions that is not using vibration data, the authors propose using Electrical Signature Analysis and Support Vector Machine (SVM) to perform fault classification for centrifugal pumps [12] by using electrical signal data (current and voltage). Their results show that the implemented method is effective in detecting and classifying main types of pump faults: vane tip fault, crack fault, leakage and cavitation. In [13] the normalized output of Electrical Signature Analysis is converted into a grayscale image which is further fed to the input of a stacked capsule autoencoder network to classify the fault.

Regarding the solutions that rely on vibration data, the literature shows an increasing interest in this direction, many complex algorithms being identified to be used for this purpose. In [14] a strategy to detect and classify faults in centrifugal pump by deriving Markov parameters containing the most significant spectral components from the covariant matrices of the time domain vibration data is proposed. The proposed method shows good accuracy in detecting cavitation, rotor and fixation faults. Most of the recent studies approach rotating machinery fault classification making use of machine learning techniques. A paper published in 2021 [15] presents a learning system that makes use of K-nearest neighbours classifier for classifying the condition of a pump based on vibration data features calculated both in time and frequency domain. Other researchers make use of CNN to solve this problem as these types of NN come with an advantage regarding the extraction of principal features of the data: CNN can choose and extract the most important features from the signal; therefore, the human expertise is not necessary anymore for this matter. [16] presents an improved CNN with an adaptable learning rate for fault diagnosis of pumps. Their strategy to use an adaptive learning rate comes with enhanced performance of the CNN, proving the effectiveness of the method for data coming from multiple sources: vibration data, pressure signal and even sound signal. Hasan et al. [17] propose a fault diagnosis framework which combines scalogram-based imaging (a time-frequency imaging technique) and Adaptive Deep CNN for fault classification of centrifugal pumps. One distinctive approach can be found in [18] where the signal data is converted into a kurtogram which is further propagated to a Convolutional Autoencoder and a CNN to extract global and local features. These features are then merged and fed to an Artificial NN for classification.

This paper deals with the fault classification of centrifugal pumps by making use of the residuum between the output and the input of Autoencoding CNN trained for abnormal behaviour detection. The residuum computed as in [2] is fed to a dimension reduction stage and to a classification stage, which constitute the research contribution of the present paper. Section II of this paper describes the physical equipment and the previously developed Autoencoding CNN. Section III presents the classification methods used, while the experimental results are described in section IV. The last section is dedicated to the conclusions of this study.

II. The Plant

A. The equipment

This paper aims at fault classification of centrifugal pumps only using vibration data. Our data used in the experiments were collected from a salt-mine plant. The plant has 11 pairs of such pumps, each of them being driven by an 18.5 kW asynchronous motor running, in permanent regime, at 3000 rpm. One such pair is shown in Fig. 1.

The maintenance team of the plant needs an alarm to be raised when one of the pumps has an abnormal behaviour and informed about the fault that could cause such a behaviour. For this reason, on each pump, a vibration sensor (accelerometer) has been mounted. Each of the accelerometers is connected to a microcontroller (Pico-PI), which is further connected in a fast RS485 network to a gateway (RPI computer), that is collecting and remotely transmitting data to the cloud where we run our analysing tools.

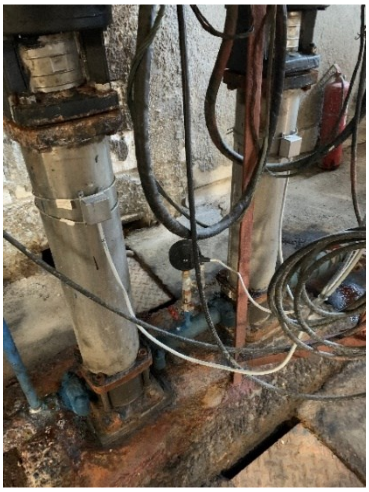

Fig. 1. One pair of pumps with sensors attached.

An extended description of the plant and the requirements for the accelerometer specifications can be found in [2]. Thus, the experiments were carried away using IIS3DWB accelerometers, which are low cost 3-axis MEMS with digital output, capable of providing output data at a rate of 26.7 KHz.

B. The system for abnormal behavior detection

For the above-mentioned plant and equipment configuration, a system for detecting abnormal behaviour using an Autoencoding Convolutional Neural Networks has been implemented in [2]. This detection method follows the steps:

- a sequence of data having a 1 second length is recorded from the accelerometer and resampled to 4000 samples. As presented and verified in [2], the band limit of 2kHz and the 1Hz spectral resolution proved to be enough for detecting the faults.
- the Fast Fourier Transform of the recorded sequence is computed.
- the first 2000 spectral components of the recorded sequence are fed into the Autoencoding CNN which is compressing and decompressing the data.
- a residuum is calculated by subtracting the values of the output spectral components from the input spectral components, resulting a vector containing the deviation of the behaviour from normal functioning signature.
- the power and the maximum amplitude of the residuum vector are computed and compared with corresponding threshold values. If any of the parameters is higher than its threshold an abnormal behaviour alarm is triggered.

The Autoencoding CNN involved in the previously mentioned detection procedure is shown in Fig. 2 and it contains 3 convolutional layers (12 filters each, with decreasing sizes 128 – 64 – 16) each of them followed by a MaxPooling layer for dimensional reduction in the encoding stage, and 3 convolutional layers (with increasing sizes 16 – 64 – 128) on the decoding stage. Between the encoding and the decoding stages there is a 50 neurons dense layer, representing the encoded representation of the data. The network has been trained with data classified by a human expert as data from normal functioning.

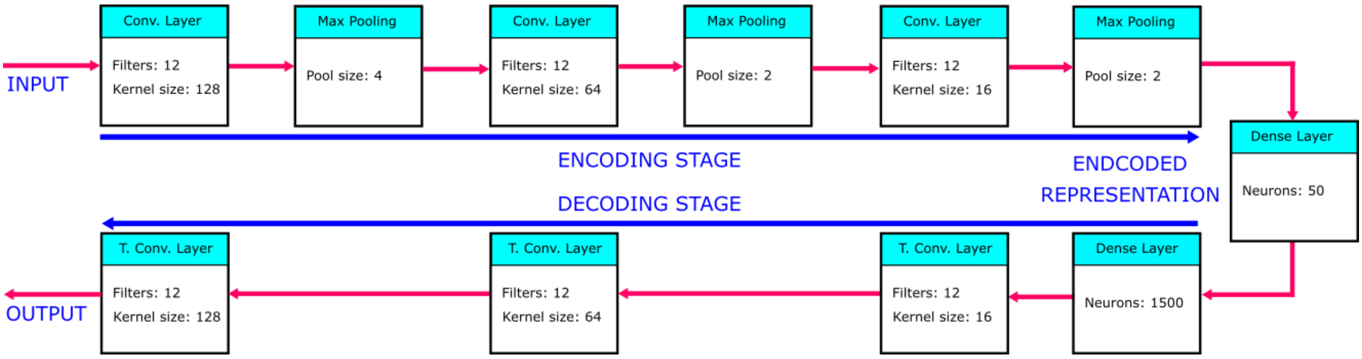

Fig. 2. Structure of the Autoencoding CNN.

The data at the input of the Autoencoding CNN can be data from normal or faulty functioning, but at the output of the network, as it has been trained with only normal data, a normal functioning signature of the pump is expected. Therefore, the residuum which is calculated by subtracting the values of the output spectral components from the input spectral components, should result in a vector containing a signature of the fault.

III. The Classification Method

The goal of this research is to implement a fault classification method for the monitored equipment that uses the residuum from the previously developed abnormal behaviour detection method. For this reason, data have been collected from the equipment in 3 fault situations (classified as such by a human expert):

1. clogged filter – 262 data sequences.
2. broken fan blade – 8094 data sequences.
3. detached rotor section (the shaft of the blade of one of the pump stages got mechanically separated from the rotor) – 8147 data sequences.

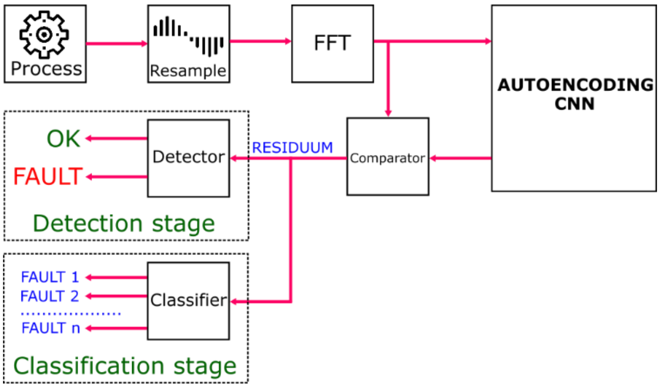

Fig. 3. Schematic representation of the detection and classification procedures.

These 3 sets of data, alongside a set of data from normal functioning, have been fed into the Autoencoding CNN, and the residuum vectors have been computed to be further used for training and validating the classifier. The flow of the detection and classification procedures, using Autoencoding CNN, is presented in Fig. 3.

A distinct problem is the choice of the method to be used for implementing the classifier. The literature presents multiple types of classifying algorithms, ranging from simple ones to complex ones that are based on machine learning techniques [19], [20]. For this case, the number of available fault examples is not enough for a statistical aproach, so the classification method is a deterministic one. When lacking *a priori* knowledge about the distribution of the data in the features' space, a non-parametric method, such as the K-nearest neighbors, is a good start. However, for this preliminary work, a better choice is to have an insight about this distribution, in a reduced dimension space. The purpose is to detect if classes are separable, if there are particular shapes of the class-specific areas and what is the position of the incipient faults within these areas. These properties lead to a simple parametric classifier, that has the advantage of reducing the effect of the measuring noise at the border regions. The parametric classifier has also the advantage of very fast computation, during the exploitation stage, when compared to the non-parametric one, because the largest part of the computation is done before this stage (off-line). However, in this application, the computation speed is not important, because the degradation of equipment is a slow process. For these reasons, before being fed to the classifier, the residuum vector goes through a dimensional reduction transformation using Principal Component Analysis (PCA). The distribution of the data is studied in this reduced space.

PCA [21] is a method for achieving dimensional reduction of data by generating a new set of data (principal components), each of them being a linear combination of the original data and orthogonally to each other. Considering m samples of dimension n, the principal components of the samples can be described by:

$$\tilde{X} = X \cdot coeff \qquad (1)$$

where $X \in \mathbb{R}^{m \times n}$ is the original data, $\tilde{X} \in \mathbb{R}^{m \times p}$ is the new data and $coeff \in \mathbb{R}^{n \times p}$ are the coefficients describing the linear combination involved in computing the first p principal components.

The $coeff$ matrix has been computed from a training dataset using the singular value decomposition (SVD) method. The training dataset is composed of the autoencoder residuum vectors of 100 data sequences for each of the three faults and 500 data sequences from normal functioning. Thus, the training dataset contains 800 residuum samples of dimension 2000 resulting in a $coeff$ matrix of size $\mathbb{R}^{2000 \times 799}$.

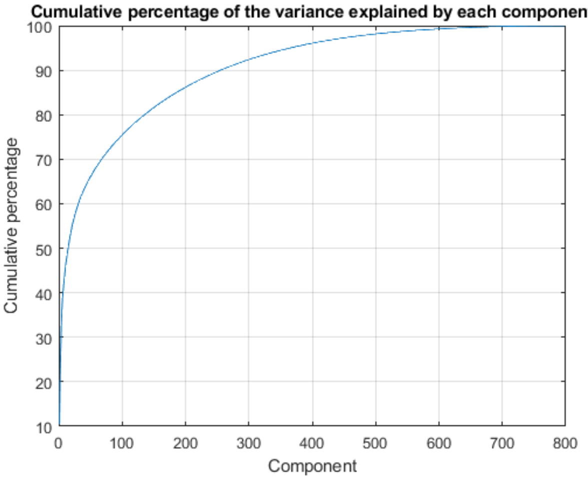

Fig. 4. Cumulative percentage of the variance explained by each component.

Using the computed $coeff$ matrix on the training dataset, the cumulative percentage of the variance explained by each component has been obtained (Fig. 4). As noticed in the cumulative sum of the percentual variance explained by each component, the first 15 principal components account for 50% of information, the first 135 principal components account for 80% of information while the first 256 components account for 90% of information.

Fig. 5 shows the first 3 principal components of the samples in the training set scattered in 3D. As noticed, there are particular patterns of the class-specific areas, easy to discriminate. The fault subsets follow distinct straight lines, while the samples from normal functioning are grouped around the coordinate system origin. Thus, the first 3 principal components carry enough information for fault identification.

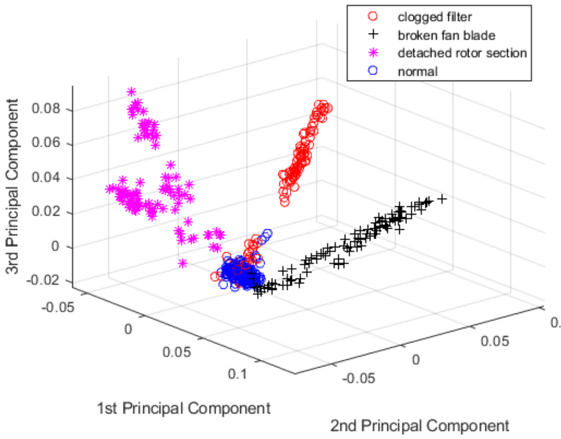

Fig. 5. The first 3 principal components of the samples in the training set

To prove this, the dimensional reduction of all available fault samples and 28051 samples of normal functioning have been computed using the given $coeff$ matrix. The first 3 principal components of the samples keep the mentioned class-specific distribution (Fig. 6). Consequently, the classification method is based on distance functions, but the shape of the class-specific areas forces two different forms of these functions: distance to the center of the class – for the normal functioning, and distance to the central line – for the fault functioning classes. The different shapes also force a mixt classification algorithm, i.e. a rule based one (decision tree): discrimination of the “normal” class first, followed by the discrimination of the fault classes, if still necessary. In all cases, the computed distance is euclidian, as no specific property was detected for a particular principal component and the extreme values of these components are comparable.

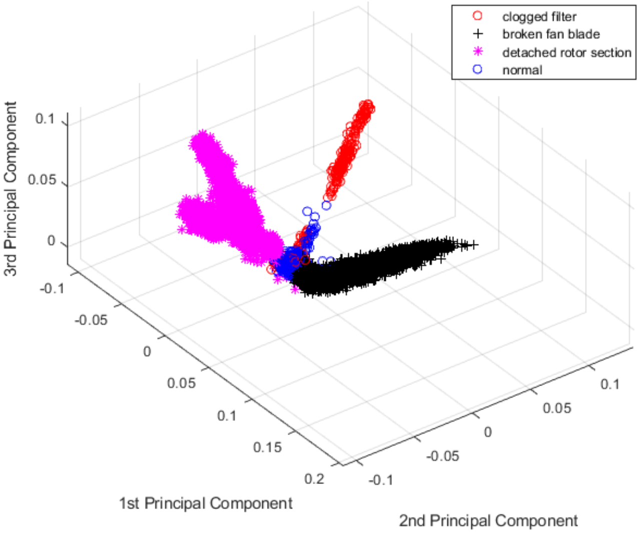

Fig. 6. The first 3 principal components of the samples in the validation set.

Aiming at this, for each fault, a linear regression to find the line that best fits the 3D samples has been computed. The vector that describes the best fitting line for the fault i is computed as:

$$\theta_i = \begin{bmatrix}\theta_{i,1}\\ \theta_{i,2}\\ \theta_{i,3}\end{bmatrix} = (Y_i^T \cdot Y_i)^{-1} \cdot Y_i^T \cdot z_i \tag{2}$$

where $Y_i \in \mathbb{R}^{100\times 3}$ is described by:

$$Y_i = \begin{bmatrix}1 & \tilde{X}_{i,1,1} & \tilde{X}_{i,1,2}\\ 1 & \tilde{X}_{i,2,1} & \tilde{X}_{i,2,2}\\ \vdots & \vdots & \vdots\\ 1 & \tilde{X}_{i,100,1} & \tilde{X}_{i,100,2}\end{bmatrix} \tag{3}$$

and $z_i \in \mathbb{R}^{100}$ by:

$$z_i = \begin{bmatrix}\tilde{X}_{i,1,3}\\ \tilde{X}_{i,2,3}\\ \vdots\\ \tilde{X}_{i,100,3}\end{bmatrix} \tag{4}$$

In equations (3) and (4) $\tilde{X}_{i,n,k}$ denotes the k principal component of the n^{th} example of fault i.

Having the line that describes the direction of each fault, for a given residuum vector, $A \in \mathbb{R}^{2000}$, the classification procedure follows the steps:

Off-line computation, during the training stage:

- The dimension of the space is reduced, according to (1) and the first 3 components are kept;
- The center of the “*normal*” class is computed as the average of the samples of this subset;

- The parameters of the central lines of the fault classes are computed according to (2), (3), (4);
- Two thresholds, denoted by $TH0$ and $TH123$ are computed. The role of $TH0$ is to allow the discrimination between the "*normal*" class and the remaining ones. If the distance between the reduced 3D vector and the center of this class is less than $TH0$, the vector is classified as "*normal*". Else, the distances to the central lines of the remaining classes are computed, in order to complete the discrimination between them. If all distances are larger than $TH123$, the vector will be classified as "*other*". The choice of $TH0$ and $TH123$ will be described at the end of this section.

On-line computation, during the exploitation stage of the classification algorithm:

Step 1: the residuum vector goes through a dimensional reduction process (PCA) by multiplying it with the previously computed $coeff$ matrix:

$$\tilde{A} = A \cdot coeff \tag{5}$$

The first 3 principal components are kept in a vector:

$$\tilde{A}_R = \begin{bmatrix} \tilde{A}_1 \\ \tilde{A}_2 \\ \tilde{A}_3 \end{bmatrix} \in \mathbb{R}^3 \tag{6}$$

Step 2: the distance between the vector $\tilde{A}_R$ and the center of the "*normal*" class is computed. **IF** the distance is lower than $TH0$ **THEN** classify as *normal* and **DONE**;

Step 3: ELSE compute the distance of the vector $\tilde{A}_R$ to each of the 3 central lines according to (7). A distance is neglected in the following steps, if it falls on the half-line, opposite to the area occupied by the corresponding subset;

$$D = \left\| \begin{bmatrix} \tilde{A}_3 \\ \tilde{A}_3 \\ \tilde{A}_3 \end{bmatrix} - \begin{bmatrix} \theta_{1,1} & \theta_{1,1} & \theta_{1,3} \\ \theta_{2,1} & \theta_{2,2} & \theta_{2,3} \\ \theta_{3,1} & \theta_{3,2} & \theta_{3,3} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ \tilde{A}_1 \\ \tilde{A}_2 \end{bmatrix} \right\| \tag{7}$$

Step 4: Get the fault candidate number, f, by determining the index of the minimum distance:

$$f = \min_i D_i \tag{8}$$

Step 5: IF $D_f < TH123$ **THEN** classify as *fault f* **ELSE** classify as *other*.

The values of the thresholds are chosen according to the risks assumed by equipment user (the client). A low value of $TH0$ determines frequent false alarms, while a large value raises the risk of missing a true fault situation. In the presented application, the maintenance team prefers to avoid too many false alarms, because some of them are the signs of incipient faults, during a slow pocesss. Accordingly, $TH0$ is chosen so as to determine less false alarms than the sum of the missed faults, when examining the training set. Regarding the value of $TH123$, we have a hint about the lower limit only, because it has to guarantee a right discrimination beyond the limits of the "*normal*" class. Its value has to be chosen larger than $TH0$. The upper limit of $TH123$ would affect the classification of vectors belonging to "other" class, occupying different class-specific areas, possibly with different shapes. Because there are no such examples yet, $TH123$ is simply chosen as above, larger than $TH0$.

IV. Experimental Results

The classification algorithm described in the previous section has been implemented in Python using the $coeff$ matrix computed for the beforementioned training dataset (autoencoder residuum vectors of 100 data sequences for each of the three faults and 500 data sequences from normal functioning). After implementing the algorithm, it has been fed with all the examples available for each fault (262 examples with clogged filter, 8094 examples with broken fan blade and 8147 examples with detached rotor section) and with 28051 examples from normal functioning to validate the accuracy of it. The two thresholds were chosen as $TH0 = 0.01$ and $TH123 = 0.05$.

The next table presents the results of the algorithm validation:

TABLE I. Confusion Matrix

Identified as	Examples labelled as			
	normal	clogged filter	broken fan blade	detached rotor section
normal	**99.70 %**	19.08 %	3.10 %	0.06 %
clogged filter	0.15 %	**79.39 %**	0 %	0.04 %
broken fan blade	< 0.01 %	0 %	**89.60 %**	0 %
detached rotor section	0.14 %	1.53 %	7.30 %	**99.90 %**
other	0 %	0 %	0 %	0 %

Further test has been performed by changing the number of the principal components used in the classification. The average percentage of correct class identification when using the first p principal components can be seen in Fig. 7. As noticed, increasing the number of principal components does not increase the performance of the classification algorithm. The random effect on this plot is produced by the small number of samples in the training set.

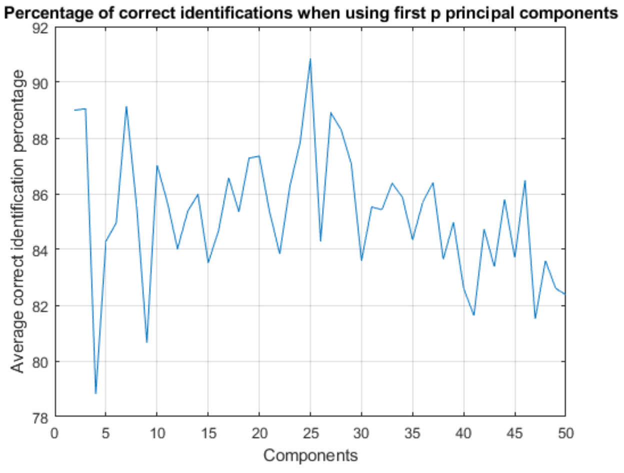

Fig. 7. Results when using the first p principal components

On the other hand, the possibility of further reducing the space dimension was also investigated. Fig. 8 presents the projection of the samples on the plane of the first 2 principal components. It proves that, for the data available up to date, 2 components are enough for a good classification. For this case, the influence of the value of $TH0$ on the classification performance was investigated. Fig. 9 presents the percentage of correct classification, for values of $TH0$ between 0.005 and 0.02. According to the preference of the equipment user, this value was chosen 0.01, while $TH123$ was kept to the value of 0.05, which is larger than $TH0$. For this threshold values, the overall resulted correct classification ability is 96.87%. The result is even better than in the case of dimension 3, because of the right choice of $TH0$ value. No classification to "*other*" was recorded, but this is due to the absence of corresponding examples in the data set and to the large value of $TH123$.

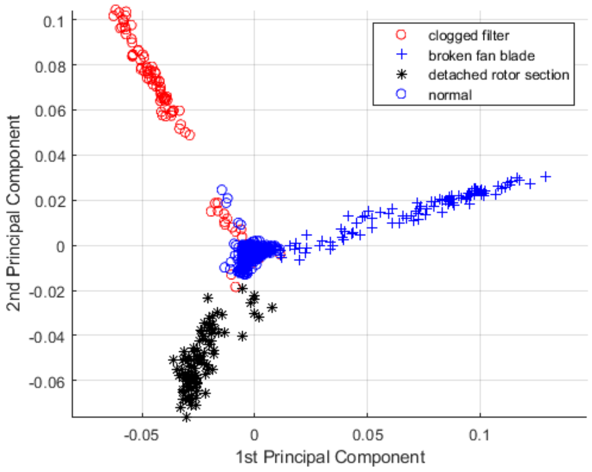

Fig. 8. The first 2 principal components of the samples in the training set.

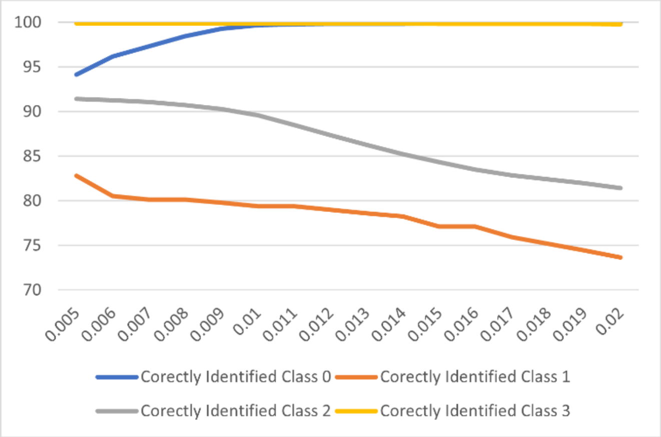

Fig. 9. Correct classification ability, function of *TH0*

Of course, the results could be further improved by setting the equipment status based on the statistical mode of the algorithm results over a certain time interval (e.g., one hour). Besides this, proven that the Autoencoding CNN residuum carries enough information for classifying the fault, a more complex classifier from the Machine Learning family could provide better performances.

For the *clogged filter* fault, a series of 788 examples collected a few days before the fault started to manifest are available. Feeding the residuum of these examples in the classification algorithm resulted in 397 examples (50 %) labelled as *normal* and 391 examples (50 %) labelled as *clogged filter*, proving this way that, with a proper statistical analysis, faults could be detected before they start physically manifest.

The classification method was further tested by continuous monitoring of the equipment. To avoid false alarms, the machine status is set based on the statistical mode of the classification algorithm results over 10 minutes. In Fig. 10 a pump status changing from 19th January 2023 can be seen.

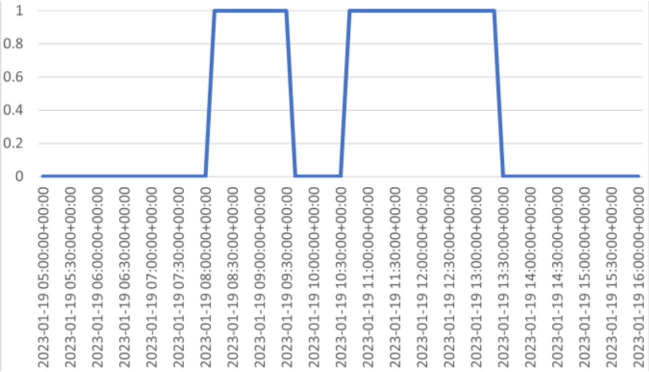

Fig. 10. Status of a pump during online monitoring.

As noticed, in the morning the machine status gets classified as 1 (*clogged filter*). Right after this the machine gets classified as 0 (*normal)* for some short time. The classification returns to *clogged filter* for a few hours before getting back to *normal* status. The maintenance team of the plant have been informed about this status change and they confirmed that, indeed, the filter of that pump got clogged with salt and has been cleaned during afternoon. After filter cleaning, the machine status returns to *normal*.

V. CONCLUSIONS

A rule-based fault classification algorithm that makes use of the residuum of a previously developed Autoencoding CNN has been implemented. No spectral analysis is used for classification in this approach. Before being fed in the classifier, the residuum vector goes through a dimensional reduction process using PCA. The use of the PCA analysis of the residuum allows a very simple form of the distance functions. More classes or more complicated class domains may require more complex distance functions, but this is not the case of the presented application. The transformation matrix of PCA has been computed based on a training set containing 100 examples from each fault and 500 examples from normal functioning of the equipment. The high dimensional space of the residuum vectors (2000) requires many observations, which were not available in the beginning of this work. As a result, the number of independent PCA components was limited to 799 (see computing the $coeff$ matrix, section III). However, the first 2 or 3 PCA components were used for classification, only. The validation step and the following tests proved that the classification ability was satisfactory, even when using the small number of examples.

The method has been tested on an extended dataset containing examples from each of the faults, alongside examples from normal functioning, obtaining an average of over 90% correct detection proving the reliability of using the autoencoder residuum as a data source for fault classification. Besides this, the classification method has proven its reliability in an industrial environment correctly classifying a detected fault. The detected fault has been confirmed by the plant's maintenance team.

Other autoencoders may be used, but this work exploited the previously developed CNN based encoder, only. The same structure of the classifier may be used when solving other problems, but the compression ratio of the CNN and the distance functions should be adapted to that particular problem, including the shape of the class domains.

The obtained results using this simple classifier confirmed that the residuum of the fault detection Autoencoding CNN provides enough information for explicitly identifying the type of fault with good accuracy. This way, there is no need to implement a separate system for fault classification as this can be done as an update to the already existing abnormal behaviour detection system. In subsequent research, the authors want to improve the classification accuracy by implementing a more complex classification algorithm based on machine learning techniques that extracts information from the same residuum. The effect of an extensive set of examples and the sensitivity to the variations of the normal functioning will be studied, as well.

ACKNOWLEDGEMENT

This work was supported by the project LINDA, SMIS130084, financed by the Romanian Ministry of Research and Innovation, Contract no. 43/221_ap2/18.06.2020.

We are also grateful to the management and operators of the salt factory, Salina SA – Târgu Ocna, Romania, for their support.

REFERENCES

[1] T. Philbeck and N. Davis, "The fourth industrial revolution: shaping a new era," *Journal of International Affairs* 72(1), 2018, pp. 17–22.

[2] I. Vasiliev, L. Frangu and M. L. Cristea, "Pump Fault Detection Using Autoencoding Neural Network," *2022 26th International Conference on System Theory, Control and Computing (ICSTCC)*, Sinaia, Romania, 2022, pp. 426-431, doi: 10.1109/ICSTCC55426.2022.9931848.

[3] T. D. Popescu, D. Aiordachioaie and M. Manolescu, "Change detection in vibration analysis — A review of problems and solutions," *2017 5th International Symposium on Electrical and Electronics Engineering (ISEEE)*, Galati, Romania, 2017, pp. 1-6, doi: 10.1109/ISEEE.2017.8170648.

[4] M. Saxena, O. Bannet, M. Gupta and R. Rajoria, "Bearing Fault Monitoring Using CWT Based Vibration Signature," *Procedia Engineering* 144, Elsevier, 2016, pp. 234-241, doi: 10.1016/j.proeng.2016.05.029.

[5] A. Hajnayeb, "Cavitation Analysis in Centrifugal Pumps Based on Vibration Bispectrum and Transfer Learning," *Shock and Vibration*, Hindawi, 2021, doi: 10.1155/2021/6988949.

[6] M. R. Nasiri, M. J. Mahjoob and H. Vahid-Alizadeh, "Vibration signature analysis for detecting cavitation in centrifugal pumps using neural networks," *2011 IEEE International Conference on Mechatronics*, Istanbul, Turkey, 2011, pp. 632-635, doi: 10.1109/ICMECH.2011.5971192.

[7] W. Zhao *et al.,* "A Novel Condition Monitoring Methodology Based on Neural Network of Pump-Turbines with Extended Operating Range," IMEKO TC10 Conference "*Testing, Diagnostics & Inspection*, Berlin, Germany, 2019.

[8] M. Valtierra-Rodriguez *et al.,* "Convolutional Neural Network and Motor Current Signature Analysis during the Transient State for Detection of Broken Rotor Bars in Induction Motors," *Sensors* 20(13), 2020, doi: 10.3390/s20133721.

[9] S. Liu *et al.,* "Bearing Fault Diagnosis Based on Improved Convolutional Deep Belief Network," *Applied Sciences* 10(18), 2020, doi: 10.3390/app10186359.

[10] E. Principi, D. Rossetti, S. Squartini and F. Piazza, "Unsupervised electric motor fault detection by using deep autoencoders," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 2, pp. 441-451, March 2019, doi: 10.1109/JAS.2019.1911393.

[11] M. P. Shah, S. N. Merchant and S. P. Awate, "Abnormality detection using deep neural networks with robust quasi-norm autoencoding and semi-supervised learning," *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, Washington, DC, USA, 2018, pp. 568-572, doi: 10.1109/ISBI.2018.8363640.

[12] Z. Araste, A. Sadighi and M. Jamimoghaddam, "Fault Diagnosis of a Centrifugal Pump Using Electrical Signature Analysis and Support Vector Machine," *Journal of Vibration Engineering & Technologies*, 2022, doi: 10.1007/s42417-022-00687-6.

[13] Z. Chang, W. Yuan, and M. Yu, "Centrifugal pump fault diagnosis method based on EAS and stacked capsule autoencoder," *Third International Conference on Electronics and Communication; Network and Computer Technology (ECNCT 2021)*, 2021, doi: 10.1117/12.2628796.

[14] J. P. S. Gonçalves, F. Fruett, J. G. Dalfré Filho and M. Giesbrecht, "Faults detection and classification in a centrifugal pump from vibration data using markov parameters," *Mechanical Systems and Signal Processing* 158, 2021, pp. 107694, doi: 10.1016/j.ymssp.2021.107694.

[15] J. Konieczny and J. Stojek, "Use of the K-Nearest Neighbour Classifier in Wear Condition Classification of a Positive Displacement Pump," *Sensors*, 21(18), 2021, doi: 10.3390/s21186247.

[16] S. Tang, Y. Zhu and S. Yuan, "An improved convolutional neural network with an adaptable learning rate towards multi-signal fault diagnosis of hydraulic piston pump," *Advanced Engineering Informatics* 50, 2021, pp. 101406, doi: 10.1016/j.aei.2021.101406.

[17] M. J. Hasan, A. Rai, Z. Ahmad, and J.-M. Kim, "A Fault Diagnosis Framework for Centrifugal Pumps by Scalogram-Based Imaging and Deep Learning," *IEEE Access* 9, 2021, pp. 58052–58066, doi: 10.1109/ACCESS.2021.3072854.

[18] A. E. Prosvirin, Z. Ahmad, and J.-M. Kim, "Global and Local Feature Extraction Using a Convolutional Autoencoder and Neural Networks for Diagnosing Centrifugal Pump Mechanical Faults," *IEEE Access*, 9, 2021, pp. 65838–65854, doi: 10.1109/ACCESS.2021.3076571.

[19] M. Kiang, "A comparative assessment of classification methods," *Decision Support Systems*, 35(4), 2003, pp. 441-454, doi: 10.1016/S0167-9236(02)00110-0.

[20] R. V. K. Reddy and U. R. Babu, "A review on classification techniques in machine learning," *International Journal of Advanced Research in Science and Engineering*, 7(3), 2018, pp. 40-47.

[21] R. Bro and A. K. Smilde, "Principal component analysis," *Anal. Methods*, 6(9), pp. 2812–2831, 2014, doi: 10.1039/C3AY41907J.