Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 72 (2018) 480–485

www.elsevier.com/locate/procedia

51st CIRP Conference on Manufacturing Systems

Self-Organizing Maps for Anomaly Localization and Predictive Maintenance in Cyber-Physical Production Systems

Alexander von Birgelen[a,*], Davide Buratti[b], Jens Mager[c], Oliver Niggemann[a,d]

[a]*Institute Industrial IT, Langenbruch 6, 32657 Lemgo, Germany*
[b]*Ocme S.r.l., Via del Popolo 20/A, 43122 Parma, Italy*
[c]*Reifenhäuser Reicofil GmbH & Co. KG, Spicher Str. 46, 53844 Troisdorf, Germany*
[d]*Fraunhofer IOSB-INA, Langenbruch 6, 32657 Lemgo, Germany*

* Corresponding author. Tel.: +49-5262-702-5934. *E-mail address:* alexander.birgelen@hs-owl.de

Abstract

Modern Cyber-Physical Production Systems provide large amounts of data such as sensor and control signals or configuration parameters. The available data enables unsupervised, data-driven solutions for model-based anomaly detection, anomaly localization and predictive maintenance: models which represent the normal behaviour of the system are learned from data. Then, live data from the system can be compared to the predictions of the model to detect faults, perform fault diagnosis and derive the overall condition of a system or its components. In this paper we use self-organizing maps for the aforementioned tasks and evaluate the presented methods on several real-world systems.

Peer-review under responsibility of the scientific committee of the 51st CIRP Conference on Manufacturing Systems.

Keywords: self-organizing map;SOM;data-driven;anomaly detection;cyber-physical production system;CPPS;anomaly localization;diagnosis;predictive maintenance

1. Introduction

Modern Cyber-Physical Production Systems (CPPS) evolve rapidly, become modular, can be parameterized and contain increasingly more sensors. This is due to increasing product variety, product complexity and pressure for efficiency in a distributed and globalized production chain [1]. This also means it becomes more and more difficult to monitor the systems. Human operators often struggle to diagnose faults or anomalous behavior in the system, leading to system break down, unexpected downtime or degradation in product quality. Maintaining the working order of a system also poses challenges: usually, components in industrial systems have defined lifespans and are replaced on a regular basis during maintenance. This is common practice but this approach has its downsides: the lifespan of components such as motors and bearings is often an estimate and the real lifespan of the component and ultimately the whole system can be influenced by environmental factors such as temperature changes, workloads and more. This can for example lead to increased cost due to components being changed too early. A far worse example is that a component reaches the end of its lifespan earlier than estimated. The system can suffer from decreased efficiency, unexpected downtime, degradation of product quality or production losses caused by such unexpected failure of a component.

A dynamic detection of a systems real condition or degradation can support experts in better planning maintenance times and avoid the aforementioned negative effects caused by system degradation. A common approach to tackle such scenario is to construct models for a given system and compare the predictions of the model to the real system [2]. Anomalous behavior is detected when the real system's behavior deviates from the model's predictions. Such models can be created in two ways: manual construction of system models by experts is usually time consuming, expensive and also difficult in today's evolving complex systems. Experts with the necessary knowledge are usually scarce and often times some of the necessary knowledge is not available at all. The second option is to learn models from data. Learned models are usually not as precise as manually constructed models but the creation of learned models requires much less effort as the heavy lifting is performed by the computer. An additional step after anomaly detection is anomaly localization: anomalous samples are presented to a reverse model to determine which signals are related to the anomaly. This provides support for plant operators and experts to restore the system to normal working order.

The data-driven condition monitoring approach presented in this paper learns models using data where the system is still in its 'new' condition where no degradation has occurred. The condition of a system is often not directly available from a single sensor and has to be derived from a multitude of signals available in the system. We use self-organizing maps (SOM)

2212-8271 © 2018 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the 51st CIRP Conference on Manufacturing Systems.
10.1016/j.procir.2018.03.150

to learn such models and use them for predictive maintenance. The information provided by the model can be used to first detect the condition or degradation of the system and second the anomaly localization supports experts so they can react to the degradation of the system and ideally repair it before it fails.

The contents of this paper are structured as follows: First, section 2 presents the SOM-based approach to detect and localize anomalies within the signal domain of a system and explains the application for condition monitoring and predictive maintenance scenarios. Third, the aforementioned approaches are explained step by step on a well known data set in section 3. Furthermore, section 3 also shows results of industry use cases. Finally, the paper is concluded in section 4.

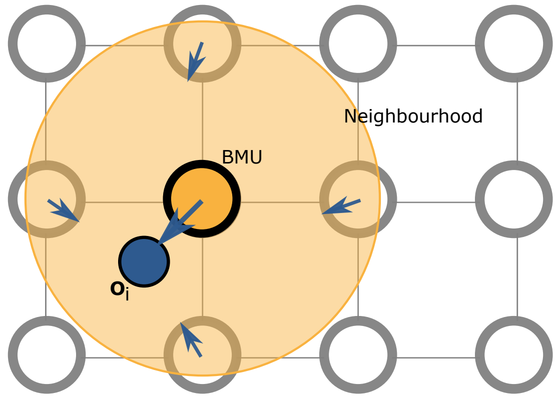

Fig. 1. Illustration of training procedure for a single sample.

2. Approach

This section introduces the background and approach for the application of SOM in predictive maintenance scenarios.

2.1. Background

The self-organizing map (SOM) [3], also referred to as self-organizing feature map or Kohonen network, is a neural network that can be associated with vector quantization, visualization and clustering but can also be used as an approach for non-linear, implicit dimensionality reduction [4].

A SOM consists of a collection of neurons which are connected in a topological arrangement which is usually a two dimensional rectangular or hexagonal grid. The input data is mapped to the neurons forming the SOM. Each neuron is essentially a weight vector of original dimensionality but provides additional information such as its coordinates within the grid. All experiments in this paper use a two dimensional, non-toroidal rectangular lattice and the Euclidean distance measure as shown in Definition 1.

Definition 1. *The $SOM = (M, G, d)$ forms a topological mapping of an input space $O \subset \mathbb{R}^m$, $m \in \mathbb{N}$ and consist of*

- *a set of neurons M.*
- *each neuron $n \in M$ has a weight vector $\mathbf{w}_n \in \mathbb{R}^m, m \in \mathbb{N}$.*
- *G is a two-dimensional rectangular lattice in which the neurons $n \in M$ are arranged.*
- *$d(\mathbf{x}, \mathbf{y})$ is the distance measure to calculate the distance between two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ which can for example be weight vectors and/or vectors in the input space. The euclidean distance is used for all models in this paper.*
- *an input sample $\mathbf{o}_i \in \mathbb{R}^m, i \in \mathbb{N}, m \in \mathbb{N}$ is mapped to the SOM through its best matching unit (BMU). The BMU is given by $bmu(\mathbf{o}_i) = \text{argmin}_{n \in M}\, d(\mathbf{o}_i, \mathbf{w}_n)$*

One way to learn a SOM from data is a random batch training approach: the initial values of the neuron's weight vectors for the training can be randomly initialized or sampled from the training data to provide a diverse starting point for the training process. Training takes place over a chosen amount of epochs. All samples from the training data are presented to the algorithm within one epoch. A best matching unit (BMU) is calculated for each input sample from the training data by finding the neuron which has the smallest distance to the sample. The BMU and all of its neighboring neurons, assigned through the topology and neighborhood radius, are shifted towards the input sample (Figure 1). Both the size of the neighborhood and strength of the shift decrease over time to help with convergence. In the end, each neuron of the SOM represents a part of the training data. Areas in the training input space with few examples are represented by few neurons of the SOM while dense areas in the input space are represented by a larger number of neurons. Usually, the number of neurons is chosen much smaller than the number of samples in the training data, effectively discretizing and reducing the training data to the most important samples.

2.2. Anomaly detection

The SOM can be used to detect anomalies by calculating the quantization error: small errors below a threshold are considered normal, while errors above are considered anomalous. Quantization error based approaches were already used for tasks such as network monitoring [5] and anomaly detection in industrial processes [6][7][8]. These works however, did not perform an anomaly localization and only [8] used the quantization error as a measure for system degradation.

The quantization error (Definition 2) of each sample is calculated by mapping it to the SOM to get its BMU. The distance of the sample to the BMU's weight vector is the quantization error.

Definition 2. *Using the notation from definition 1, the quantization error* **qe** *of an input sample $\mathbf{o}_i \in \mathbb{R}^m, i \in \mathbb{N}$ is given by the distance of the input sample to its BMU of the SOM:* $\mathbf{qe} = d(\mathbf{o}_i, bmu(\mathbf{o}_i))$.

The quantization errors for data that is not anomalous are usually greater than 0 due to the discretization of the SOM. A threshold for the quantization error above which an input sample is classified as anomalous is required. Manual selection of the threshold works but is usually unfeasible for practical applications. It is far more convenient to estimate the threshold from data: the quantization errors of the training data can be seen as a probability distribution and quantiles can be used to retrieve the threshold for the anomaly detection. The quantile can be adjusted and we will use the parameter τ with $\tau \in \mathbb{R}$ and $0.0 \leq \tau \leq 1.0$ within this paper. This can be adjusted to optimize the outcome of the anomaly detection: when labels are present τ can be used to fine tune the anomaly detection score. When the training data is perfect, meaning it contains only normal behavior, no sampling errors, no glitches in the sensors and no noise, then a τ of 1.0 is fine as this results in the maximum error for the threshold. However, training data is never perfect when working with real systems and data might contain a small portion of samples affected by noise and/or other effects. The maximum error might be too large to effectively find anomalies. Setting τ to a value slightly smaller than 1.0 can increase the true positive rate of the anomaly detection at the cost of some false positives, depending on the use case and desired outcome.

2.3. Anomaly localization

An additional step after the anomaly detection is to calculate the signal or sensor which is most likely to cause the anomaly. Anomaly localization is performed after an anomaly is detected: the observation found to be anomalous is fed through a reverse model to obtain the expected values for the signals. The deviations from the expected values can then be used to identify the signals related to the anomaly. The weight vector of each neuron of the SOM has the same dimensions as the input data and each element of the weight vector contains the value of its corresponding signal. Once an anomaly is detected, the input sample is again mapped to the SOM to retrieve the BMU. Now, the distance of each signal to the weight vector is calculated and the resulting signals and their distances are sorted in a descending order according to their distance. Since real world systems usually provide a large number of signals it is necessary to reduce the number of displayed signals. Therefore only the first n signals are displayed giving plant experts and operators a starting point to locate the anomaly and possible fault in the system and ultimately restore the system's normal working order. For the experiments in this paper we only consider the signal with the largest deviation (n=1).

2.4. Predictive maintenance

Both techniques described here can be combined for unsupervised condition monitoring and predictive maintenance: The training data is selected such that it only contains the 'new' condition of the system. A SOM is learned with the data and an error threshold on the training data is estimated using the aforementioned method. Now, new data is mapped to the SOM and the deviation is calculated. The system degrades over time as it is used and its condition should drift farther and farther away from its new condition. Therefore, the deviations from the model is expected to increase over the lifetime of the system. Initially, the raw value produced by the model is only a number which indicates some level of degradation. Experts can refine the predictions of the model by looking the deviation produced by the system at the end of its life-time. This deviation can be used to define a more concise value for the prediction. The deviation of a system at the end of its lifetime could be interpreted as 0% life and a deviation of 0 as 100% life. Knowing the real condition of the system gives experts an edge in planning maintenance windows: maintenance windows can be shifted according to the system condition and maintenance can be performed at times where it least interferes with system operation. Also, the localization provides better insight on the anomalies and experts can hone in on the specific part of the system and restore the system to a working state. The full procedure is visualized in Figure 2.

3. Experiments

The methods presented in this paper were applied to a variety of data and experimental results from three different condition monitoring applications are presented in this section.

3.1. Applying anomaly detection and localization for predictive maintenance

The data was generated by the NSF I/UCR Center for Intelligent Maintenance Systems with support from Rexnord Corp. in Milwaukee, WI [9]. It contains three run to failure experiments. Four bearings were installed on a rotating shaft. The rotation speed was kept constant at 2000 RPM by an AC motor coupled to the shaft via rub belts. A radial load of 6000 lbs is applied onto the shaft and bearing by a spring mechanism. All bearings are force lubricated. Rexnord ZA-2115 double row bearings were installed on the shaft. PCB 353B33 High Sensitivity Quartz ICP accelerometers were installed on the bearing housing (two accelerometers for each bearing [x- and y-axes] for data set 1, one accelerometer for each bearing for data sets 2 and 3). All failures occurred after exceeding designed life time of the bearing which is more than 100 million revolutions. Each data set consists of individual files that are 1-second vibration signal snapshots recorded at specific intervals. Each file consists of 20,480 points with the sampling rate set at 20 kHz. The time when the file was recorded is known from the file name.

The algorithms introduced before are now applied to these data sets: first train a SOM to represent the 'new' condition of the system. Second, the quantization error is used to derive the overall condition of the system. Third, the algorithm for anomaly localization is used to find which of the sensors, or here which bearing, is responsible for the degradation of the condition. Data set number two will be used to explain the process in detail, before results for all three data sets will be given. The data set contains run-to-failure data, so in the beginning the bearings are still considered new and the experiment runs until a failure occurs. The data set contains 984 files with 20480 observations each. The first 10% of the data are chosen to represent the 'new' condition and a SOM is trained on this chosen data. For the second data set this means that SOM is trained on the first 90 files with a total of 1.843.200 samples. The data is normalized to a range between 1 and 0. The size of the SOM was set to 50x50 neurons and training took place over 100 epochs. These are the standard settings for the training algorithm; other sizes work too and the number of epochs and size of the SOM are not super critical for the data sets in this paper. A larger SOM represents the training data in more detail compared to a smaller one at the cost of computation efforts during the training and search for BMU later on. The quantization error of the training data is calculated and τ is set to 0.9999 for the estimation of the training error. Deviations smaller than the estimated threshold will be considered normal (Figure 3).

Now the full data set is compared to the learned SOM and the quantization error is calculated for every sample. The maximum deviation found in each file is selected and is plotted in Figure 4 in combination with the time of the file to visualize the

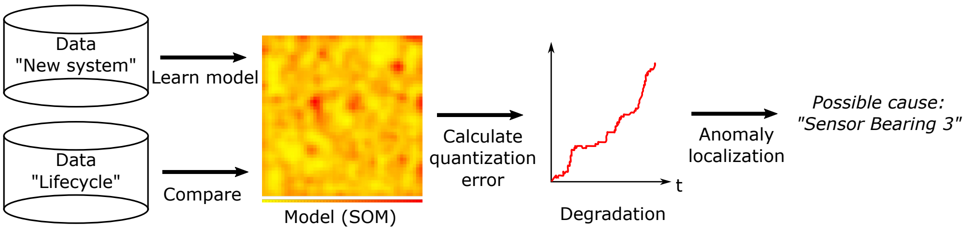

Fig. 2. Illustration of the full procedure.

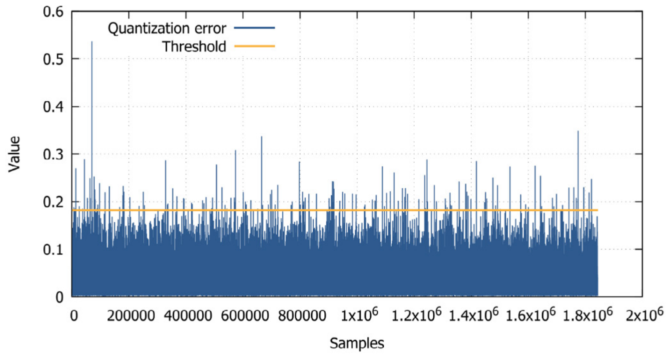

Fig. 3. Estimated threshold of the training data for the second bearing data set.

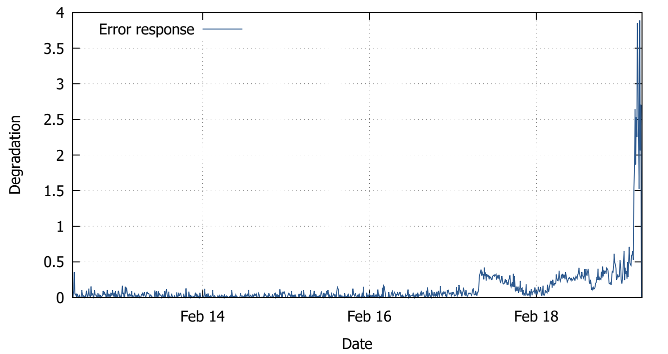

Fig. 4. Quantization error as a measure of degradation for the second bearing data set.

degradation.

According to the documentation, an outer race failure occurred in bearing 1 at the end of test. A first spike in the degradation seems to occur on the 17th of February. The anomaly localization indicates that the deviation on the 17th of February, starting around 7:42 in the morning, is caused by S0, which first translates to the first column in the data. The first column is from the vibration sensor located at bearing 1. Total failure starts on the 19th of February at around 4:02 in the morning. The deviation drastically rises until bearing 1 fails and stopping the experiment at around 6:02 on the 19th of February, which is also reflected by the anomaly localization. The model was able to detect the failure in bearing 1 almost two days before it failed completely. This time span can be used by the service team to prepare and react to system failures before complete failure occurs.

Figure 5 provides more insight of the anomaly localization process: every file of the bearing data contains 20480 observations and the anomaly localization is performed on every observation and the occurrences are counted and plotted in form of filled curves. When the system is working normal, we expect an equal distribution across all signals reflecting the normal noise

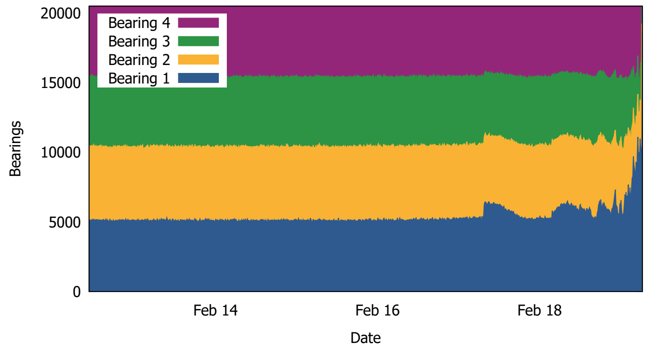

Fig. 5. Plot of the anomaly localization of the second bearing data set.

of the data. As the system degrades, the signals related to the degradation should be drawn more often than the other signals. This is confirmed by the result: at the beginning all bearings are equally drawn by the anomaly localization. Once degradation becomes visible, bearing 1 is drawn more often than the other bearings which marks it as the likely cause of the anomaly. Figure 5 nicely shows the degradation of bearing 1 in the second bearing data set. The curve follows the detected degradation shown in Figure 4.

The same procedure can be applied to the first and third data set. The degradation for the bearings in the first data set are depicted in Figure 6. The anomaly localization alternates between bearings 3 and 4, starting from the 17th of November 14:22 as shown in Figure 7. During the final hours of the experiments, starting on the 25th of November on 13:17, the anomaly localization indicates bearing 3 as most likely cause for the degradation of the system.

The degradation for the bearings in the third data set is shown in Figure 8 and ends with failure of bearing 3, which starts to become obvious on the 16th of April around 20:12 before the experiment ends on the 18th of April at 02:42 in the morning (Figure 9).

The degradation of the system was successfully detected in advance of total failure in all three data sets and even the exact bearing was identified in the anomaly localization.

3.2. Use cases in industrial plants

The previous section showed that the aforementioned algorithms work on a small system under very controlled circumstances. The presented method is not limited to vibration sensors and can work with any numerical signals and can be applied to a variety of systems. The following applies the methods to two examples from industry: in both cases the goal was to derive the condition of an important component of the system from a variety of already existing senors and signals such as electrical currents, torque and temperatures. The results shown

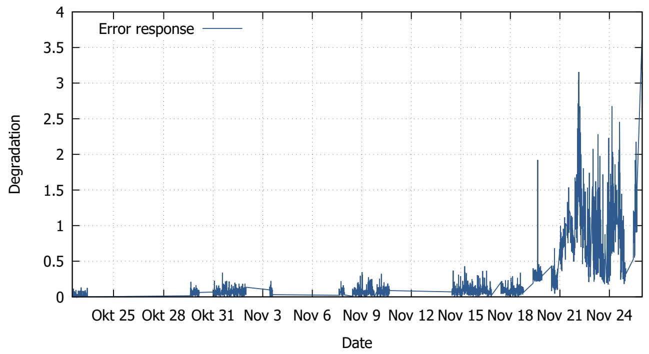

Fig. 6. Degradation for the first bearing data set.

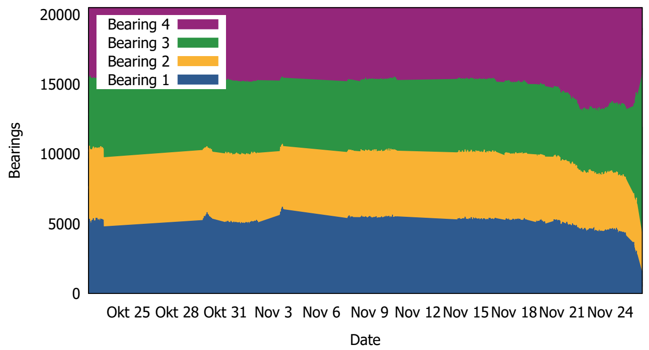

Fig. 7. Anomaly localization for the first bearing data set.

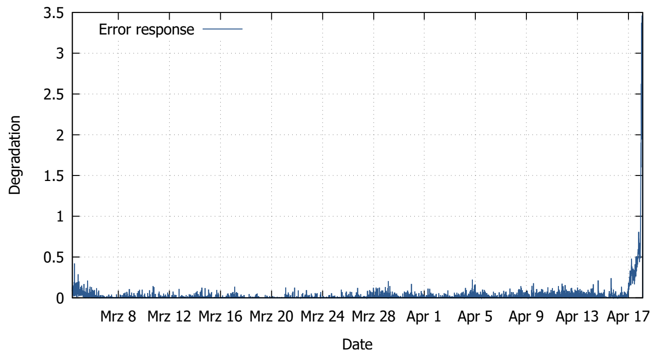

Fig. 8. Degradation for the third bearing data set.

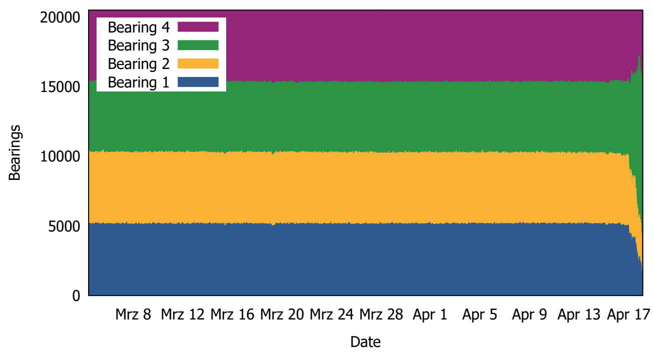

Fig. 9. Anomaly localization for the third bearing data set.

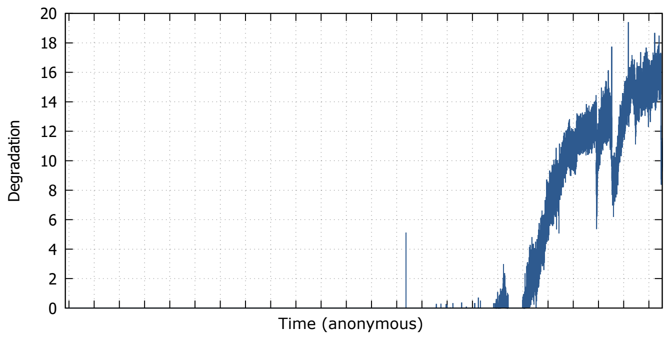

Fig. 10. Degradation of component at Reicofil

in this paper are anonymised and details about exact signal names are excluded to protect the interests of the involved companies.

3.2.1. Reifenhäuser Reicofil GmbH & Co. KG

The use case from Reicofil focuses on the prediction of the condition of an important component within their composite lines for non-woven materials [10]. The condition of this component is important for the function of the plant and the resulting product quality. Data for 8 run-to-failure experiments were provided and 8 features related to the component were selected. Training and prediction data were selected using the leave-one-out method: data from the component under test were selected as the target for the prediction. A set amount of data of all other components were selected and combined to serve as training data for the 'new' condition. Again, a SOM was trained on a training data to represent the 'new' condition. The degradation of the component under test was calculated and visualized. This procedure was repeated for all 8 data sets to get a prediction of the degradation for all components. Figure 10 shows the result for one of the 8 data sets. The prediction worked for all cases which were labeled with a certain type of wear by experts. Furthermore, one of the components did not show signs of wear according to experts which was also confirmed by the model.

3.2.2. Ocme S.r.l.

The Vega shrink-wrapper from OCME is deployed in large production lines in the food and beverage industry [11]. The machine groups loose bottles or cans into set package sizes, wraps them in plastic film and then heat-shrinks the plastic film to combine them into a package. The plastic film is fed into the machine from large spools and is then cut to the length needed to wrap the film around a pack of goods. The cutting assembly is an important component of the machine to meet the high availability target. Therefore, the blade needs to be set-up and maintained properly. Furthermore, the blade can not be inspected visually during operation due to the blade being enclosed in a metal housing and its fast rotation speed. Monitoring the cutting blades degradation will increase the machines reliability and reduce unexpected downtime caused by failed cuts. For this use case, we still work on obtaining full run-to-failure data. The following shows results from two snippets of data comparing two different new blades to a completely worn out blade. Figure 11 shows the comparison from a new cutting blade and data from a worn out cutting blade, recorded on the same machine. A small portion was randomly drawn from the training data to validate the learned model. The deviations de-

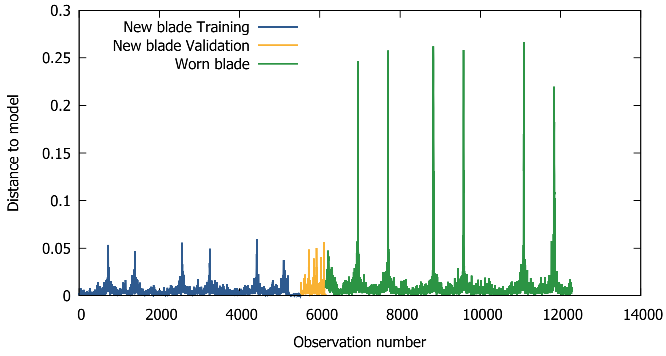

Fig. 11. Comparison between a new and a worn blade.

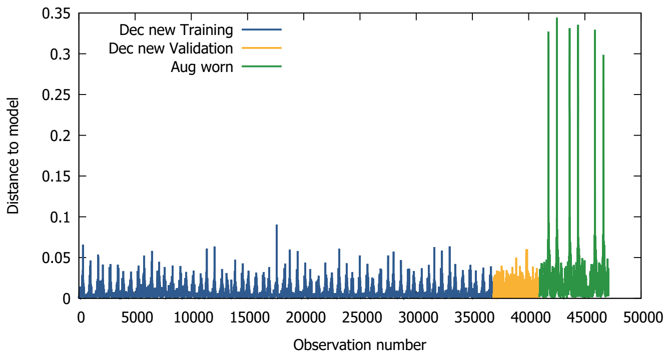

Fig. 12. Comparison between a new and a worn blade. The new blade data was recorded on a different machine to test transfer ability.

tected for the validation data should be similar to the training data. The prediction with the worn blade shows a higher average error and especially the peaks in the error response, which happen when the cutting blade makes contact with the plastic film, are a good indication for the wear of the cutting blade. Figure 12 shows data of the second new cutting blade, recorded on a different machine, and the worn out cutting blade to test the ability to transfer the learned model. Again, a small portion was randomly drawn from the training data to validate the learned model. The detection of the blade degradation also indicating that models can be transferred between machines.

4. Conclusion

This paper presented an unsupervised approach based on SOM for data-driven anomaly detection and localization in CPPS and its application for condition monitoring and predictive maintenance. Components degrade over time as part of their normal life-cycle. Oftentimes parts are replaced on fixed time intervals during planned maintenance windows. The major disadvantages of this approach are that components are replaced too often or worse that a component fails and is replaced after the fact. The degradation (and ultimately failure) of components lead to reduced efficiency, reduced product quality or unexpected downtime of the system.

These problems can be counteracted by data-driven condition monitoring approaches: data provided by the system is used train model to represent the 'normal' or 'new' behavior of important components or the complete system. Usually, the condition is not available directly through a sensor and has to be derived from a multitude of available signals. SOMs are a powerful model which can be used for this purpose. Deviations from the model are calculated and used as a measure for the degradation of a component. The quantization error provides information about the severity of the degradation. This degradation of the condition can be monitored over time and for example fitted with a regression line to estimate the future development of the degradation. An additional step is the anomaly localization which uses the SOM as a reverse model to calculate which sensors might cause the degradation. Both of these steps provide important information to the service teams of the plant: the information of the degradation in general can be integrated into the planning of future maintenance windows or extra maintenance can be performed at a time where it least interferes with machine operation. The anomaly localization provides information to better locate the origin of the degradation giving experts a head start on the analysis. The algorithms are explained and validated on the well-known bearing data set [9]. Furthermore, results from two industrial use cases are included to show the the algorithms working on more complex, real-world data.

Future research includes the estimation of the remaining useful life of the system. One of the simplest methods is to perform a regression, using the quantization error as dependent variable and the time as independent variable. However, such a simple model seems to be not suitable for the estimation of remaining useful life and different methods, such as holt-winters-forecasting, are currently being evaluated for this purpose.

Acknowledgments

IMPROVE has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 678867.

References

[1] Factories of the Future: Multi-Annual Roadmap for the contractual PPP under HORIZON 2020. Luxembourg: European Union; 2013.
[2] Niggemann, O., Lohweg, V.. On the diagnosis of cyber-physical production systems: State-of-the-art and research agenda. In: Twenty-Ninth AAAI Conference on Artificial Intelligence. 2015,.
[3] Kohonen, T.. The self-organizing map. In: Proceedings of the IEEE; vol. 78. 1990,.
[4] Yin, H.. The Self-Organizing Maps: Background, Theories, Extensions and Applications. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-540-78293-3; 2008, p. 715–762.
[5] Kumpulainen, P., Hätönen, K.. Local anomaly detection for mobile network monitoring. Inf Sci 2008;178(20):3840–3859.
[6] Simula, O., Kangas, J.. Process monitoring and visualisation using self-organizing maps. 1995.
[7] Frey, C.. Monitoring of complex industrial processes based on self-organizing maps and watershed transformations. In: Industrial Technology (ICIT), 2012 IEEE International Conference on. 2012,.
[8] Tian, J., Azarian, M.H., Pecht, M.. Anomaly detection using self-organizing maps-based k-nearest neighbor algorithm. Second European Conference of the Prognostics and Health Management Society 2014 2014;.
[9] Lee, J., Qiu, H., Yu, G., Lin, J., Rexnord Technical Services, . Bearing data set. Tech. Rep.; IMS, University of Cincinnati; NASA Ames Research Center, Moffett Field, CA; 2007. URL: `https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/`.
[10] Reifenhäuser Reicofil GmbH & Co. KG, . Reifenhäuser reicofil - composite lines. 2017. URL: `http://www.reicofil.com/en/pages/composite_lines`.
[11] OCME, . Shrink-wrap packers vega. 2017. URL: `http://www.ocme.com/en/our-solutions/secondary-packaging/vega`.