CNN Based Predictive Maintanance of Generalized Rotational Equipment

Oleg Rudenko[1], Oleksandr Bezsonov[1], Andrzej Szajna[2], Stanislav Piskunov[3]

[1]*Kharkiv National University of Radioelectronics, Nauky ave., 14, Kharkiv, 61166, Ukraine*
[2]*Uniwersytet Zielonogórski, Zielona Gora,Licealna str., 9, 65417, Poland*
[3]*Ivan Kozhedub National Air Force University, Sumska str., 77-79, Kharkiv, 61023, Ukraine*

Abstract

Predictive maintenance (PdM) is a technique that uses performance indicators of some unit to detect anomalies in its behavior. In data measured by sensors it is possible to find unusual patterns, such as growing engine vibration or high power consumption. Usually, malfunctions are preceded by some unusual behavior of the machine. The huge amount of the existing sensors is generating a huge volume of data. Machine learning algorithms can gather and analyze patterns from the data and create models to estimate the necessary machine health metrics. This paper examines the structure of the convolutional neural network (CNN) and its training methods for solving the problem of aggregate maintenance prediction in combination with the method of converting one-dimensional data into images. The data transformation step is very important for further use of CNN. A generalized model of rotating equipment was used to evaluate the proposed approach. The data was preprocessed, transformed into an image, and fed to a customized classifier. The simulation results showed that the combination of CNN with the data dimensionality enhancement method outperformed traditional machine learning methods (random forest, support vector machine), and forecasting methods based on MLP, LSTM and 1-d CNN neural-networks.

Keywords

Predictive Maintenance, Deep Learning, Convolutional Neural Networks, Gramian angular field, Signal Encoding.

1. Introduction

Predictive maintenance is a powerful tool that enables companies to predict and prevent equipment failure, minimizing downtime, and increasing productivity. With the advent of deep learning techniques, predictive maintenance has become more accurate and efficient, leading to better equipment performance and reduced costs. Predicting the behavior of complex machines is an important scientific and technical problem that allows predicting the behavior of production processes and significantly reducing risks in environmental, economic, social and other systems. The relevance of the task of predicting possible malfunctions of complex systems is especially growing recently. This is due to the availability of powerful computing tools for collecting and processing information [1–5].

The development of prognostics as a science in recent decades has led to the creation of many models and methods, procedures, techniques of forecasting, unequaled in their importance. According to estimates of foreign and domestic experts in prognostication, there are already more than a hundred

COLINS-2023: 7th International Conference on Computational Linguistics and Intelligent Systems, April 20–21, 2023, Kharkiv, Ukraine
EMAIL: oleh.rudenko@nure.ua (O. Rudenko); oleksandr.bezsonov@nure.ua (O. Bezsonov); a.szajna@dtpoland.com (A.Szajna); piskunoff@ukr.net (S.Piskunov)
ORCID: 0000-0003-0859-2015 (O. Rudenko); 0000-0001-6104-4275 (O. Bezsonov); 0000-0003-4975-0534 (A.Szajna); 0000-0002-4685-527X (S.Piskunov)

methods of forecasting, in connection with which the task of choosing methods that would provide adequate forecasts for the investigated processes or systems arises.

Until recently, the statistical approach was the main one in solving the forecasting problem. Within the framework of statistical models, the tasks of forecasting, finding hidden periodicities in data, analyzing dependencies, assessing risks in decision-making, and others are solved. A general disadvantage of statistical models is the difficulty of choosing the type of model and selecting its parameters.

It should be noted that strict statistical assumptions regarding the properties of time series, which describe signals from machine sensors, significantly limit the possibilities of classical forecasting methods. In addition, when using a statistical approach, one of the main requirements for time series is its stationarity, which consists in the fact that the distribution of its values is invariant with respect to the moment in time for which it was constructed. It also significantly limits the capabilities of classical forecasting methods.

An alternative to statistical methods can be the methods of computational intelligence, among which artificial neural networks (ANNs) should be included in the first place [1-2]. Being universal approximators, some types of ANNs allow to restore with a given accuracy any arbitrarily complex continuous nonlinear function, using the representation of the approximated function in the form of a neural network formed by neurons, the parameters of which are determined by its training. The ability of a neural network for versatile processing of information stems from its ability to generalize and highlight hidden dependencies between input and output data. A significant advantage of ANNs is that they are capable of learning and generalizing accumulated knowledge. The use of these methods to solve forecasting tasks is due to the presence of complex regularities, a priori and current uncertainty, non-stationarity, emissions, etc. [6–8].

Intelligent forecasting methods built in this way, like classical ones, require the creation of a mathematical model, the quality of which determines the accuracy of forecasting. It should be noted that the same criteria are used for building the model (choosing its structure and estimating parameters) that characterize the accuracy of forecasting [9-10].

When studying time series, it is almost impossible to get an absolutely accurate forecast, for this reason, it is considered an important task to evaluate various forecasting models from the point of view of certain quality criteria. At the same time, the reliability of the selected forecasting model is assessed by periodic comparison of the actual and forecasted values of the series. When solving practical problems, the analysis of prediction error or prediction accuracy is considered more significant [11].

In this article, we will explore how encoding signals by Gramian Angular Fields and further recognition by Convolutional Neural Networks (CNNs) can enhance predictive maintenance. Gramian Angular Fields (GAF) is a powerful tool for encoding time-series data into image-like representations, allowing it to be processed by convolutional neural networks [12-16]. GAF is a matrix that represents the phase relationships between the values of the time series. It can be thought of as a signal's "footprint in the phase space," and it contains information about the signal's frequency and amplitude characteristics. GAF has several advantages over traditional time-series data processing techniques. For example, GAF is able to preserve the structure of the time series, allowing it to capture complex patterns and trends. Additionally, GAF is computationally efficient, making it suitable for large datasets.

The aim of this work is to develop a structure for predictive maintenance of equipment using a combination of the GAF method used to transform a signal into an image and a convolutional neural network CNN. The use of CNN is a significant advantage over traditional machine learning methods, as it allows the use of fairly well-researched and proven neural network models. Since the health of the system is constantly analyzed in this approach, a deviation from the standard behavior immediately indicates that the machine needs to be called for unscheduled maintenance. The paper shows that the proposed approach makes it possible to detect anomalous data sequences and give more accurate recommendations on the need to perform system maintenance. Modern production systems contain a significant set of sensors that collect a huge amount of data from which the combination of GAF and CNN methods can extract hidden patterns and get better results compared to traditional statistical forecasting methods. A small amount of work in this area is due to the fact that previously similar methods machine learning has not been used with 1D data from analog sensors.

2. Related Works

Predictive maintenance is a crucial aspect of modern industry that aims to reduce downtime, optimize maintenance schedules, and enhance overall productivity. Recent advancements in deep learning have led to the development of several innovative PdM techniques based on machine learning algorithms, such as CNNs. In this section, we review some of the related works in the field of PdM using CNNs and highlight the contributions of this research in the context of encoding time-series data into images.

In recent years, CNNs have emerged as a powerful tool for PdM due to their ability to learn complex features from raw data. Several studies have investigated the use of CNNs for fault detection in industrial systems. For example, Xiang et al. [17] proposed a deep learning approach based on CNNs for the early detection of gear faults in wind turbines. Similarly, Guo et al. [18] presented a CNN-based method for detecting bearing faults in rotating machinery.

One of the key challenges in PdM is the representation of time-series data, which is typically high-dimensional and complex. To overcome this challenge, several studies have explored the use of image-based approaches to represent time-series data. One such approach is the GAF and it has been used for encoding time-series data in several domains, such as finance [14] and health [19].

In the context of PdM, Choudhary et al. [20] proposed a CNN model for the fault diagnosis of rolling element bearings. They showed that their approach achieved better results than traditional feature-based methods. Lee et al. used GAF-based CNNs for the rolling element fault diagnosis under various operating and noisy conditions [21]. They demonstrated that their method outperformed other state-of-the-art approaches.

In summary, the proposed method in this article builds on the mentioned works by exploring the use of GAF-based CNNs for predictive maintenance in industrial systems. The contribution of this research lies in the evaluation of the proposed method on a dataset generated with using of the rotating machine emulator and the comparison with other state-of-the-art techniques.

3. Proposed Method

To apply GAF to predictive maintenance, we start by collecting data from sensors on the equipment. This data can include vibration, temperature, pressure, and other relevant signals. We then use GAF to encode this data into image-like representations, which can be further processed by CNNs.

CNNs are a powerful class of deep learning models that are capable of learning complex patterns and structures in images. In predictive maintenance, CNNs can be trained to recognize specific patterns in the GAF-encoded time series data, such as changes in vibration amplitude or frequency. By detecting these patterns, CNNs can predict when equipment failure is likely to occur, enabling maintenance personnel to take action before the failure occurs.

To train the CNN, we start by splitting the data into training and validation sets. We then use the training data to train the CNN to recognize patterns in the GAF-encoded time series data. Once the CNN is trained, we use the validation data to evaluate its performance and make any necessary adjustments.

Once the CNN is trained and validated, it can be used to predict when equipment failure is likely to occur. This can be done by feeding new GAF-encoded time series data into the CNN and using its output to make a prediction about the equipment's health. If the CNN predicts that equipment failure is likely, maintenance personnel can take action to prevent the failure from occurring. Overall method description is presented at Fig.1.

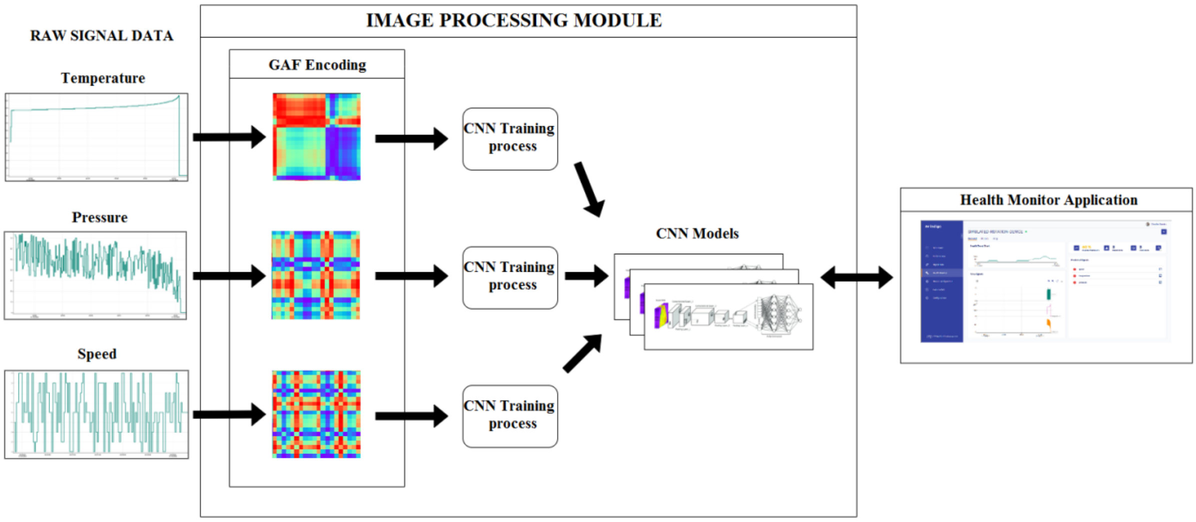

Figure 1: GAF based predictive maintanance

4. Conversion of one-dimensional signals into images

The Gramian Angular Field (GAF) visualization is an effective encoding method of one-dimensional signals from sensors in the form of images. This method has been proposed Wang and Oates in [12].This method has become very popular due to the fact that it allows the use of well-established convolutional neural networks for signal processing, which turn out to be much more effective than traditional methods for processing one-dimensional signals.

Converting 1D signals to images is one option for data dimensionality enhancement as it transitions from 1D data to 2D or even 3D. Dimensionalization is an important step when using CNN models. In [13], an approach called Gramian Angular Field (GAF) is presented for encoding time series into images to improve classification. The GAF method uses a matrix based on polar coordinates to encode time series into an image, since they have the ability to store information about temporal correlation, unlike Cartesian coordinates [13]. The algorithm for converting signals to GAF images contains the following steps:

1) Initial time series is normalized and reduced to values in the interval [−1, 1] using the expression

$$\bar{x}_i = \frac{(x_i - \max(X)) + (x_i - \min(X))}{\max(X) - (x_i - \min(X))}, \quad (1)$$

where $-1 \le \bar{x}_i \le 1$ the normalized value of each original signal value x_i .

2) The second step in building time series using the GAF method is to present normalized time series $\bar{X}$ in polar coordinates, which are calculated by finding the angular cosine of each normalized value and the timestamp, represented as a radius, using the expressions:

$$\theta = \arccos(\bar{x}_i) \quad (2)$$

$$r = \frac{t_i}{M}, \quad (3)$$

where θ – the value of the time series in the format of polar coordinates; t_i - timestamp of time series data; M is some constant used to stabilize the space of the polar coordinate system.

3) After obtaining the polar coordinates for each value of the time series, the trigonometric sum is calculated to find the spatial correlation between the polar points as follows:

$$GAF = \left[\cos(\theta_i + \theta_j)\right].$$

GAF has the following several advantages:

- according to[14-16], temporal correlation is represented by a superposition of time intervals;
- in the GAF matrix, the main diagonal contains the original value and angle information;

- the GAF method preserves the temporal correlation of the time series input data, which is necessary to build an efficient forecasting system.

In this regard, GAF provides high-quality images for CNN, which allows to reveal complex relationships between various states of a rotating machine.

Fig.2. shows a graphical representation of the conversion of time series to GAF images

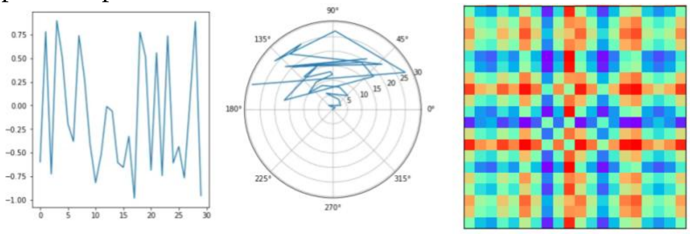

Figure 2: Graphical representation of the conversion of time series to GAF images

5. Convolutional Neural Network (CNN)

The Convolution Neural Network (CNN) was first proposed in [22, 23] as a development of the neocognitron model designed for efficient image recognition. Subsequently, based on the CNN, R-CNN networks (Regions With CNNs) were built to apply CNN to the object detection problem. R-CNN creates a bounding box for each object in an image, or a suggestion of regions, using a selective search process. Fast R-CNN, which has increased the performance of R-CNN, performs the classification of the objects of each region along with tighter bounding boxes. The next Faster R-CNN network improved the mechanism for generating candidate regions used in it by calculating regions not from the original image, but from the feature map obtained from the CNN. To do this, a module called the Region Proposal Network (RPN) was added. Finally, the Mask R-CNN network develops the Faster R-CNN architecture by adding another branch that predicts the position of the mask covering the found object, and thus solves the instance segmentation problem. When receiving an image, the network issues objects (bbox), bounding boxes, classes (class) and masks (mask).

It should be noted that Mask R-CNN is the fastest network at the moment.

5.1. Structure of a Convolutional Neural Network

Initially, the structure of a convolutional neural network was created taking into account the structural features of some parts of the human brain responsible for vision. The development of such networks is based on three mechanisms:

- local perception;
- formation of layers in the form of a set of feature maps (shared weights);
- subsampling.

By local perception it is meant that not the entire image, but only some part of it, comes to the input of the neuron. This allows to save the configuration of the image when moving from layer to layer.

The idea of shared weights implies that a small set of weights is applied to a large number of links, i.e. each area of the image into which it is divided will be treated by the same set of weights. With such an artificially created weight restriction, the generalization property of the network improves.

The CNN consists of layers of convolution, subsampling and layers of a fully connected neural network.

5.2. Convolutional Neural Network Layers

CNNs get their name from the "convolution" operator. The main purpose of convolution in case of CNN is to extract elements from the input image. Convolution preserves the spatial relationships between pixels by learning the features of an image using small squares of the input.

Each neuron in the convolutional layer plane receives its inputs from some area of the previous layer (local receptive field), that is, the input image of the previous layer is scanned with a small window and passed through a set of weights, and the result is mapped to the corresponding neuron in the convolutional layer.

The subsampling layer scales down the planes by locally averaging the neuron output values. Thus, a hierarchical organization is achieved. Subsequent layers extract more general characteristics that are less dependent on image distortion.

Each convolutional layer is followed by a subsampling layer, or computational layer, which performs image dimensionality reduction by locally averaging the neuron output values (Fig. 3). In the architecture of a convolutional network, it is generally accepted that the presence of a feature is more important than information about its location. Therefore, from several neighboring neurons in the feature map, the maximum is selected and its value is considered to be one neuron in the feature map of a lower dimension. The difference between the subsampling layer and the convolution layer is that in the latter, the regions of neighboring neurons overlap, which does not happen in the subsampling layer.

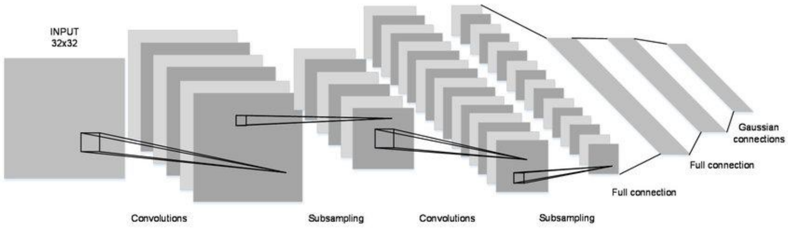

Figure 3: Convolutional Neural Network Structure

Thus, the CNN is built by alternating layers of convolution and subsampling. At the output of the network, several layers of a fully connected neural network are usually installed, at the input of which finite feature maps are fed. Each neuron of this layer is a perceptron that has a non-linear activation function.

5.3. Convolutional Neural Network Parameter Tuning Methods

To train convolutional neural networks, both the standard backpropagation method and its various methods can be used modifications. Let's consider one of them.

Consider training a multilayer perceptron (MP) with M inputs and L outputs, S layers, the number of neurons in the output layer P, hidden - N, all neurons of which have an activation function of the form

$$f(x) = \frac{1}{1+e^{-\alpha x}}$$

Let us introduce the following notation:

$W^{*s}(k) = \left(W_1^{*s}(k), W_2^{*s}(k), ..., W_p^{*s}(k)\right)^T$ – matrix of optimal weights of the output layer PxL;

$w_i^{*s}(k) = \left(w_{i1}^{*s}(k), w_{i2}^{*s}(k), ..., w_{iL}^{*s}(k)\right)^T$ – is the vector of optimal weights of the i-th neuron of the output layer Px1;

$W^{s}(k) = \left(W_1^{s}(k), W_2^{s}(k), ..., W_p^{s}(k)\right)^T$ – matrix of estimates of the weights of the output layer of the MP PxL;

$w_i^{s}(k) = \left(w_{i1}^{s}(k), w_{i2}^{s}(k), ..., w_{iP}^{s}(k)\right)^T$ – is the vector of estimates of the weights of the i-th neuron of the output layer Px1;

$W^{*i}(k) = \left(W_1^{*i}(k), W_2^{*i}(k), ..., W_p^{*i}(k)\right)^T$ – is the matrix of optimal weights of neurons of the i-th hidden layer ($i = 1, ..., S-1.$) NxP;

$w_i^{*i}(k) = \left(w_{i1}^{*i}(k), w_{i2}^{*i}(k), ..., w_{iN}^{*i}(k)\right)^T$ – vector of optimal weights of neurons of the i-th hidden layer Nx1;

$F^s(x, w^s) = (f_1^s(w_1^{Ts}(k)x(k)) \quad f_2^s(w_2^{Ts}(k)x(k)) \quad ... f_L^s(w_L^{Ts}(k)x(k)))^T$ – is the vector of activation functions of neurons in the output layer Lx1;

$\nabla f(x,w) = diag[\nabla f(w_1^T(k), x(k)) \quad \nabla f(w_2^T(k), x(k)) \quad ... \nabla f(w_L^T(k), x(k))]$ – is the matrix of derivatives of the activation functions of the LxL layer;

$\nabla f(w_i^T(k), x(k))$ – is the first derivative of the activation function of the i-th neuron.

Then the output signals and the approximation error can be represented as follows:

- required vector of output signals

$$y^*(k) = W^{*s}(k)F^s(x, w^{*s}(k)) + \xi_1(k), \tag{4}$$

- real vector of output signals

$$y(k) = W^s(k)F\left(x, w^s(k)\right) + \xi_2(k); \tag{5}$$

- error

$$e(k) = y^*(k) - y(k) + \xi(k) = \tilde{\theta}^T(\mathrm{k})F\left(x, w(k)\right) + W^{*s}(k)(k)\tilde{F}\left(x, w(k)\right) + \xi_1(k) + \xi_2(k); \tag{6}$$

$i = 1, ..., S-1.$

Where $\tilde{\theta}(\mathrm{k}) = W^{*s}(k) - W^s(k)$ is the learning error matrix PxL;

$\tilde{F}(k) = F\left(x, w^{*s}(k)\right) - F\left(x, w^s(k)\right)$ is the error vector of activation functions;

$\xi_1(k)$, $\xi_2(k)$ are the error vector of the output signal and the neural network approximation, respectively.

With these notations in mind, the gradient procedures for learning the weight matrices of the output and i-th hidden layer of the MP, linearizing the quadratic functional and representing the error backpropagation procedure, can be written as follows:

$$W^s(k) = W^s(k-1) - \gamma_{W^s}(k)\frac{1}{\left\|F\left(x, w^s(k-1)\right)\right\|^2}\frac{\partial E(k)}{\partial W^s(k-1)} = W^s(k-1) + \gamma_{W^s}(k)\frac{F\left(x, w^s(k-1)\right)e^T(k)}{\left\|F\left(x, w^s(k-1)\right)\right\|^2}; \tag{7}$$

$$W^i(k) = W^i(k-1) - \gamma_{W^i}(k)\frac{1}{\left\|\nabla f\left(x(k), W^i(k-1)\right)x(k)\right\|^2}\frac{\partial E(k)}{\partial W^i(k-1)} = \tag{8}$$

$$= W^i(k-1) + \gamma_{W^i}(k)\frac{\nabla f\left(x(k), w^i(k-1)\right)e^T(k)x^T(k)}{\left\|\nabla f\left(x(k), W^i(k-1)\right)x(k)\right\|^2},$$

$i = 1, ..., S-1,$

those is a matrix version of the Kaczmarz (Widrow-Hoff) procedure.

Here $\gamma_{W^s}(k)$ And $\gamma_{W^i}(k)$, $i = 1, ..., S-1$ – learning coefficients.

To increase the computational stability of (7), (8), they can be modified by introducing regularization parameters into them, i.e.

$$W^s(k) = W^s(k-1) + \gamma_{W^s}(k)\frac{F\left(x, w^s(k-1)\right)e^T(k)}{\beta_{W^s}(k) + \left\|F\left(x, w^s(k-1)\right)\right\|^2}; \tag{9}$$

$$W^i(k) = W^i(k-1) + \gamma_{W^i}(k)\frac{\nabla f\left(x(k), w^i(k-1)\right)e(k)x^T(k)}{\beta_{W^i}(k) + \left\|\nabla f\left(x, W^i(k-1)\right)x(k)\right\|^2}, i \neq S, \tag{10}$$

where $\beta_{W^s}(k)$, $\beta_{W^i}(k), i = 1, ..., S-1$ are the regularization parameters.

However, it should be noted that if the statistical properties of the noise $\xi_1(k)$ And $\xi_2(k)$ are not known, but it is known that they are limited in amplitude, then procedures (7)-(10) should be modified by using dead zones in them, as was done for the case of linear Adalina, for example, as follows:

$$W^s(k) = W^s(k-1) + \gamma_{W^s}(k) \frac{\alpha_{W^s}(k) F\left(x, w^s(k-1)\right) e^T(k)}{\beta_{W^s}(k) + \left\| F\left(x, w^s(k-1)\right) \right\|^2}, \tag{11}$$

where

$$\alpha_{W^s}(k) = \begin{cases} g\left(e(k), \Delta_{W^s}(k)\right), \ if \ \|e(k)\| > \Delta_{W^s}(k); \\ 0, \ if \ \|e(k)\| \le \Delta_{W^s}(k). \end{cases} \tag{12}$$

Since when calculating the estimates of the matrix W the derivative of the activation function is used, then the dead zone parameter must contain the coefficient α, defining the slope of the sigmoid

$$W^i(k) = W^i(k-1) + \gamma_{W^i}(k), \frac{\alpha_{W^i}(k) \nabla f\left(x(k), W^s(k-1)\right) e(k) x^T(k)}{\beta_{W^i}(k) + \left\| \nabla f\left(x, W^s(k-1)\right) x(k) \right\|^2}, \tag{13}$$

where

$$\alpha_{W^i}(k) = \begin{cases} g\left(e(k), \Delta_{W^i}(k)\right), \ if \ \dfrac{\nabla f_{\min}}{\alpha} \|e(k)\| > \Delta_{W^i}(k); \\ 0, \ if \ \dfrac{\nabla f_{\min}}{\alpha} \|e(k)\| \le \Delta_{W^i}(k), \end{cases} \tag{14}$$

where $\nabla f_{\min} = \min\left[\nabla f_1(k), \nabla f_2(k), ..., \nabla f_L(k)\right] > 0.$

Setting the dead zones for procedures (11) and (13) can be done as follows:

$$\Delta_{W^s}(k) = \Delta_{W^s}(k-1) + \frac{\alpha_{W^s}(k) \|e(k-1)\|}{\beta_{W^s}(k-1) + \left\| F\left(x, W^s(k-1)\right) \right\|^2}; \tag{15}$$

$$\Delta_{W^i}(k) = \Delta_{W^i}(k-1) + \frac{\alpha_{W^i}(k) \|e(k-1)\|}{\beta_{W^i}(k-1) + \left\| \nabla f\left(x, W^i(k-1)\right) x(k) \right\|^2}, i \ne S. \tag{16}$$

6. Experiments.

All experiments were performed with using rotating machine emulator [24] for generating needed data. The practical application of maintenance solutions for complex systems is associated with the presence of a complex infrastructure for remote monitoring of the state of a given system. The quality of forecasting directly depends on the availability of significant amounts of telemetry and service records to failure, on the basis of which it is possible to build models of system degradation. In this case, failure prediction can be made both on the basis of historical and newly collected data.

Obtaining real data sets for solving the problem of building a system degradation model is a very difficult task. This is due to the reluctance of commercial organizations to post telemetry data from their facilities for public use. It should be noted that there are some synthetic datasets, but none of them fully correspond to the approach in which irregular real-time data streams from sensors are used for condition prediction or anomaly detection [25].

In this paper, we use the data generation method presented in [24]. It is a model of a production environment with versatile rotating equipment (pumps, turbines, compressors and various motors). This model can be used to generate arbitrarily large synthetic data for training a neural network model.

The generated data set is a combination of system maintenance records and telemetry. In this case, the data read from the sensor, together with the parameters describing the current state of the machine, are transferred from the emulated system to some cloud service that predicts the state of this system.

The messages sent to the cloud service also contain information about system failures, indicating the exact times and types of failures. The purpose of preventive maintenance is to prevent such events by giving preventive alarms to maintenance personnel so that appropriate preventive actions can be taken. Each such message contains fields with timestamps, information related to danger level, event code and machine ID.

The rotating machine emulator used in this paper generates a telemetry stream that include vibration level, temperature, pressure, and other relevant signals like timestamp and machine ID.

It should be noted that for more efficient modeling of a real system, it is necessary to take into account the change in the parameters of a real system over time. To do this, the model uses the generalized equation of equipment wear at the macro level over time

$$w = Ae^{B(t)}. \quad (17)$$

Equation (17) can be rewritten as

$$h(t) = 1 - h_0 - e^{\alpha t^{\beta}},$$

where h_0 – initial wear; α and β are the coefficients that determine the rate of degradation of the system.

In the investigated emulator of rotating machines are also simulated weather conditions, location and some characteristics of the operator. Operating conditions undoubtedly affect the performance and life of the equipment, so they must be taken into account in predictive maintenance.

In most cases, the approaching breakdown of rotating equipment is manifested in a gradual decrease in useful work and a change in the level of side characteristics. If the simulated machine has pressure as a "useful signal", then a machine running at the same speed will generate less and less pressure and yet generate more heat and/or vibration.

The general idea behind preventive maintenance is that different types of impending breakdowns show up differently over time, and that such patterns can be explored with enough data collected.

Various failure modes due to unknown physical damage are modeled by overlaying independent time-varying health characteristics $h_i(t)$ with some randomly chosen failure occurrence parameters (α_i and β_i). The work of each N simulated devices starts with random non-zero initial wear h_0 to account for changes due to age, manufacturing defects and other unknown factors.

Each device is modeled on a random sequence of time-varying operating parameters, generating telemetry signals containing all observed performance and measurements that are affected by the current values of health indicators. If any of the health indicators falls below the threshold (a value dependent on the failure mode), a corresponding entry is added to the unit's maintenance log. Non-Gaussian noise is added to all generated telemetry data to simulate random processes that affect a real system.

At any given time, the rotary machine has a certain desired speed, measured in revolutions per minute (RPM). This speed can vary (i.e. the engine can start, stop, gradually accelerate or decelerate) or be relatively constant. Smooth changes of states in time are modeled using cubic or linear interpolation.

Instantaneous temperature and pressure can be modeled using some function $F(t)$

$$F(t) = f(\omega, F(t-1), h^*(t), a^*(t)),$$

where – ω current settings, $h^*(t)$ – current wear indicator; $a^*(t)$ – environmental conditions.

The analysis of vibration and acoustic signals is a necessary tool for monitoring the condition of rotating equipment. Vibration monitoring uses non-invasively installed sensors that collect a one-dimensional analog signal representing vibration at a specific location and periodically transmit it to a cloud service for predictive maintenance. In this case, the frequencies of the harmonic components of the vibration signal are modeled as multiples of the main frequency of the machine speed. A continuous vibration or audio signal can be modeled using spectral simulation synthesis.

An example of generated data is given at Fig. 4. At this picture there are 3 channels of data: temperature, pressure and rotation speed. As it is seen from the picture length of timeframe between critical failures varies according to (17).

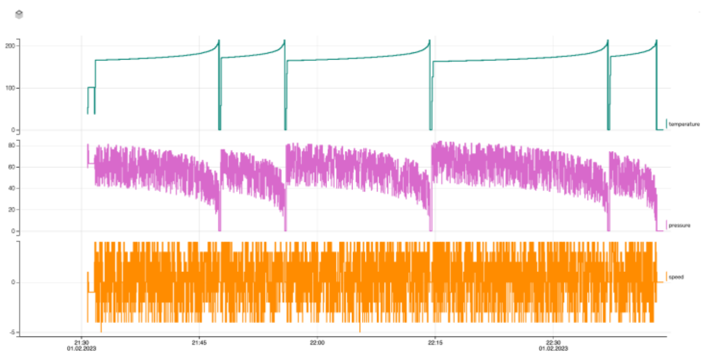

Figure 4: Generated data

The difference in encoding different system states with GAF method considering temperature data channel is presented at Fig .5.

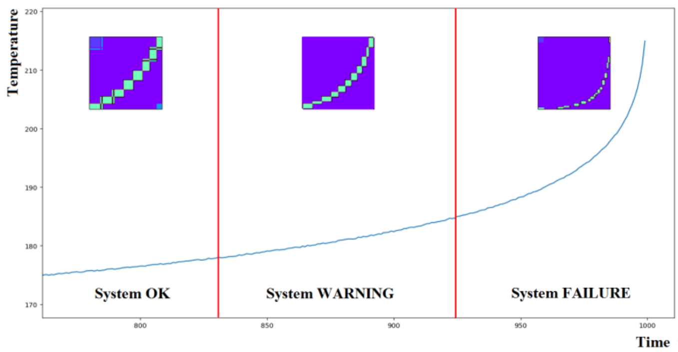

Figure 5: Example figure

For the purpose of testing the considered method about 10k of rotating machine cycles were emulated. For encoding data from emulated sensors into GAF images a “sliding window” method was used. The size of each window was 5 minutes and delta was 1 minute. It means that each minute of the simulated time last 5 minutes of data in each data channel was encoded into 64x64x3 images. Resulting images were collected into 3 dataset: training (about 30k images), validation (about 10k images) and testing (5k images). For each data channel a separate CNN model was trained (Fig. 1). The overall system health can be estimated as an average of each model prediction. Experiments show that in 93% cases the considered method was able to predict properly approaching system failure within 30 min time period.

Accuracy of the proposed method was compared with other time-series predictive solutions based on machine-learning methods like random forest, support vector machine, and neural-network methods based on a multilayer perceptron (MLP), long short-term memory (LSTM) and 1-d CNN neural networks. Results of the comparison are presented at Table 1.

Table 1
Comparison Results

Neural Network Type	Input data dimension	RMSE	Forecast accuracy, %		
			5 min	30 min	1h
SVM	1-d	3.489	83	80	75
Random Forest	1-d	2.934	85	81	79
MLP	1-d	2.631	86	81	74
CNN	1-d	1.661	91	87	82
LSTM	1-d	1.489	94	89	85
GAF + CNN	2-d	1.233	97	93	91

7. Conclusions

In conclusion, encoding signals by Gramian Angular Fields and further recognition by Convolutional Neural Networks is a powerful technique for enhancing predictive maintenance. By using GAF to encode time-series data into image-like representations and processing it with CNNs, we can detect complex patterns and structures in the data, enabling us to predict equipment failure with greater accuracy and efficiency. As a result, predictive maintenance can help companies reduce costs, increase productivity, and improve equipment performance, leading to better business outcomes. Simulation results on rotating machine emulator showed that in 93% cases proposed method generates correct predictions of the incoming in 30 min system failure. In real life it means significant saving of resources and decreasing of overall system downtime.

8. References

[1] B. Bowerman, R. O'Connell, A. Koehler,. Forecasting: methods and applications. 4th ed. Belmont, CA.: Thomson Brooks, 2005, 384 p.

[2] J. D. Kelleher, B. McNamee, A. D'Arcy, Fundamentals of Machine Learning for Predictive Data Analytics, second edition: Algorithms, Worked Examples, and Case Studies (2nd ed.), The MIT Press, 2020, 856 p.

[3] R. J. Hyndman, A. B. Koehler, Another look at measures of forecast accuracy, Int. J. of Forecasting, 22 (4), 2006, pp. 679–688.

[4] J.G.D. Gooijer, R.J. Hyndman, 25 Years of Time Series Forecasting, Int. Journal of Forecasting, 22(4), 2006, pp. 679–688.

[5] H. Allende, C. Moraga, R. Salas, Artificial neural networks in time series forecasting: a comparative analysis, Kybernetika, 6, 2002, pp. 685–707.

[6] L.J. Cao, F.E.H. Tay, Support Vector Machine with Adaptive Parameters in Financial Time Series Forecasting, IEEE Transaction on Neural Networks, 14(6), 2003, pp. 1506-1518.

[7] F.E.H. Tay, L. Cao, Application of support vector machines in financial time series forecasting, Omega, 29, 2001, pp. 309–317.

[8] P. Goodwin, R. Lawton, On the asymmetry of the symmetric MAPE, Int. J. of Forecasting, 15, 1999, pp. 405–408.

[9] O.G. Rudenko, E.V. Bodyanskyi, Artificial neural networks: Teaching. book, Kharkiv: SMIT Company LLC, 2006. – 404 p.

[10] E.V. Bodyansky, O.G. Rudenko, Artificial neural networks: architectures, training, applications, Kharkiv: TELETECH, 2004, 369 p.

[11] A.B. Koehler, The asymmetry of the sAPE measure and other comments on the M3-competition, Int. J. of Forecasting, 17, (2001), pp. 570–574.

[12] Z. Wang, T. Oates, Encoding time series as images for visual inspection and classification using tiled convolutional neural networks, In: Work Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.

[13] E. Guresen, G. Kayakutlu, T.U. Daim, Using artificial neural network models in stock market index prediction, Expert Systems with Applications, 38(8), 2011, pp. 10389–10397.

[14] S. Barra, S. M. Carta, A. Corriga, A. S. Podda, and D. R. Recupero, "Deep learning and time series-to-image encoding for financial forecasting," in IEEE/CAA Journal of Automatica Sinica, volume 7, 3, May 2020, pp. 683-692, doi: 10.1109/JAS.2020.1003132.

[15] G. Mart´ınez-Arellano, G. Terrazas, S. Ratchev, Tool wear classification using time series imaging and deep learning, The International Journal of Advanced Manufacturing Technology, 104, (2019), pp. 3647–3662, doi: 10.1007/s00170-019-04090-6

[16] K. S. Kiangala, Z. Wang, An Effective Predictive Maintenance Framework for Conveyor Motors Using Dual Time-Series Imaging and Convolutional Neural Network in an Industry 4.0

Environment, in IEEE Access, volume 8, 2020, pp. 121033-121049, doi: 10.1109/ACCESS.2020.3006788.
[17] L. Xiang, P. Wang, X. Yang, A. Hu, H. Su, Fault detection of wind turbine based on SCADA data analysis using CNN and LSTM with attention mechanism, Measurement, volume 175, 2021, 109094, doi: 10.1016/j.measurement.2021.109094.
[18] Z. Guo, M. Yang, X. Huang, Bearing fault diagnosis based on speed signal and CNN model, Energy Reports, volume 8, Supplement 13, 2022, pp. 904-913, doi: 10.1016/j.egyr.2022.08.041.
[19] H. Xu, J. Li, H. Yuan, Q. Liu, et al., Human Activity Recognition Based on Gramian Angular Field and Deep Convolutional Neural Network, IEEE Access, 8, 199393-199405, doi: 10.1109/ACCESS.2020.3032699.
[20] A. Choudhary, T. Mian, S. Fatima, Convolutional neural network based bearing fault diagnosis of rotating machine using thermal images, Measurement, volume 176, 2021, 109196, doi: 10.1016/j.measurement.2021.109196.
[21] C.Y. Lee, G.L. Zhuo, T.A. Le, A Robust Deep Neural Network for Rolling Element Fault Diagnosis under Various Operating and Noisy Conditions, Sensors (Basel), volume 22(13):4705, 2022, Jun 22;. doi: 10.3390/s22134705.
[22] Y. LeCun, A theoretical framework for backpropagation, Proc. of IEEE, 1998, pp.21-28.
[23] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient Based Learning Applied to Document Recognition, Proc. of IEEE, volume 86, 11, 1998, pp. 2278-2324.
[24] https://github.com/Azure/AI-PredictiveMaintenance
[25] https://github.com/Microsoft/SQL-Server-R-Services-Samples/tree/master/PredictiveMaintanenceModelingGuide