INTELLIGENT MONITORING AND FAULT PREDICTION SYSTEM FOR DRIVES OF AN INDUSTRIAL, 6-AXIS ROBOT

[1]PAWEŁ PASŁAWSKI, [2]MICHAŁ PIETRZAK, [3]PAWEŁ PRZEJCZOWSKI, [4]JAN PRZEPIÓRA, [5]KRZYSZTOF LALIK

AGH University of Science and Technology in Kraków
E-mail: [1]paslawsk@student.agh.edu.pl, [2]mpietrz@student.agh.edu.pl, [3]przejczo@student.agh.edu.pl, [4]przepial@student.agh.edu.pl, [5]klalik@agh.edu.pl

Abstract - This paper presents the results of an intelligent system for monitoring and prediction of defects in the drives of an industrial robot. A neural algorithm based on a convolution neural network has been implemented to detect robot malfunctions. The algorithm receives input data thanks to sensors mounted in the robot's drives. The collected data - current and torque - are stored in the cloud. This makes it possible to apply advanced data processing algorithms and, as a result, to detect early failures and wear of the drive. The article presents the architecture of the implemented neural network, how the data is prepared for learning and the results of learning the algorithm.

Keywords - Artificial Intelligence, Predictive Maintenance, Industry 4.0, Convolutional Neural Network.

I. INTRODUCTION

The primary objective of the work was to develop a damage prediction system for a six-axis robot. Robots of this type are widely used in industry. Anthropomorphic robots are an integral part of many process lines and are used for a variety of tasks such as moving parts, assembling components and welding therefore their failures cause millions of dollars in losses due to downtime. Therefore, early detection of failures before they occur can reduce the amount of downtime and allow better planning of maintenance work. Predictive maintenance is one of the parts of Industry 4.0, and its main goal is to eliminate the negative effects of failures in the form of downtime, as well as to improve the monitoring of machine health.
Predictive maintenance is a proactive approach to equipment and facility maintenance that uses data analytics and machine learning techniques to predict when maintenance will be needed [1,2,3]. By predicting when equipment is likely to fail, maintenance can be scheduled in advance, reducing the risk of equipment failure and increasing overall facility performance. Predictive maintenance also helps reduce maintenance costs because it is only performed when it is actually needed, rather than on a fixed schedule. Some common techniques used in predictive maintenance include vibration analysis, oil analysis and thermal imaging [4,5]. These techniques enable maintenance teams to identify potential problems before they become major issues, ensuring that equipment is kept in optimal condition and reduces the need for costly repairs. Industry 4.0, also known as the fourth industrial revolution, is a term referring to the current trend of automation and data sharing in manufacturing technologies, including the Internet of Things (IoT), artificial intelligence (AI) and machine learning [6,7,8]. It represents a major change in the way manufacturing processes are organized and executed, with the goal of increasing efficiency, productivity and flexibility. One of the key aspects of Industry 4.0 is the integration of advanced technologies such as IoT, AI and machine learning into manufacturing processes [9-14]. This allows data to be collected and analyzed from various sources, including sensors, machines and products, in real time. This data can then be used to optimize production, improve quality and reduce waste and downtime. Another important aspect of Industry 4.0 is the use of advanced automation technologies, such as robotics, to increase the speed and flexibility of production. This allows manufacturers to respond quickly to changing market conditions and customer demands, while reducing the need for manual labor.Overall, Industry 4.0 represents a significant change in the way manufacturing is carried out, aiming to increase efficiency, productivity and flexibility through the use of advanced technologies. It has the potential to revolutionize the way goods are produced and consumed, and is expected to have a significant impact on the global economy in the coming years [15,16,17].

II. DATA ACQUISITION

A FANUC 200ic industrial robot was selected as the test subject. The robot moved a metal block between two points in a loop. The real system is shown in Fig. 1.

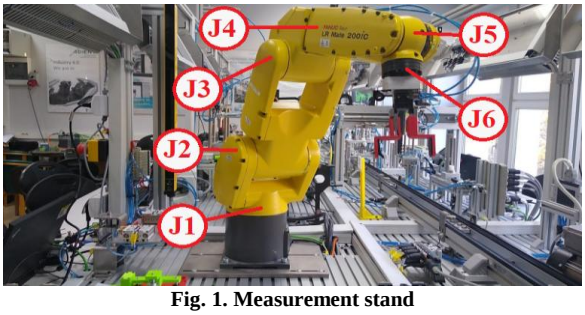


Fig. 1. Measurement stand

During the robot's cycle, torque and current measurements were taken on all the robot's drives, as well as the temperature of the two most heavily loaded drives. The actual values of torque, current, and temperature will be sent to the cloud for storage, allowing advanced neural algorithms to be utilized. An additional functionality is the possibility to view the measured values in real time. Fig. 2 presents the graphical interface prepared for this purpose. Using HoloLens augmented reality glasses, the user can see the current values of the measured signals and their waveforms from the last 5 minutes of operation.



Fig. 2. Graphical interface

The motion trajectory was chosen so that each joint made a rotation while moving the object. This made it possible to obtain visible changes in current and torque values during the robot's operation. The temperature value, even during prolonged operation, remained constant, so this data was not used to teach the network. Fig. 3 and 4 present examples of torque and current waveforms on the robot's drive 1.

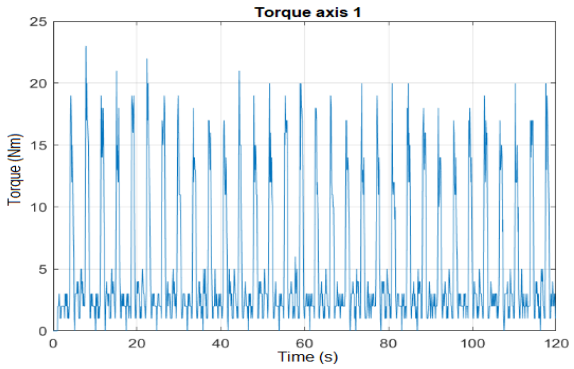


Fig. 3. Torque on the robot's drive

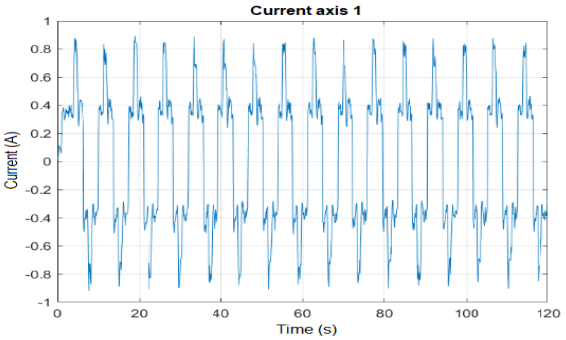


Fig. 4. Current on the robot's drive

The next step, after taking data from the robot's correct cycles, was to simulate an emergency situation. Such an event could be the wear of the drive gears, which causes an increase in the resistance to movement. The result is an increase in the torque required, and thus an increase in the current drawn by the drive. To simulate such a situation, an additional mass was loaded on the robot arm at the wrist. With this modification, the same operation cycles were repeated and another portion of data was collected for learning.

III. CONVOLUTIONAL NEURAL NETWORK FRAMEWORK

For solving the problem posed, a convolutional neural network (CNN) was chosen, which is a type of artificial neural network specifically designed to process data with a grid-like structure, such as images. It is particularly effective in image classification and has been widely used in various applications, including object recognition, face recognition and medical imaging [18,19,20].

The CNN framework consists of multiple layers of artificial neurons, each of which is responsible for processing a small portion of the input data. Neurons in the first layer receive the raw input data and pass it through a series of splicing and combining operations that extract features from the data and reduce its dimensionality. The output of these operations is passed through additional layers of neurons that further refine and classify features. Finally, the output of the CNN is a prediction or classification based on the input data.One of the key advantages of the CNN platform is its ability to learn and adapt to new data. By adjusting the weights and biases of the neurons in each layer, the CNN is able to learn complex patterns and relationships in the data, improving performance over time. This learning capability makes CNNs a powerful tool for a wide range of applications in fields such as computer vision, natural language processing and speech recognition [21-26].

The network used has a sequential structure. It has six hidden layers, consisting of alternating convolutional layers with an increasing number of nodes and Max Pooling layers. A rectified linear unit (ReLU) was used as the activation function.

```
model = tf.keras.models.Sequential([tf.keras.layers.Conv2D(16,(3,3),activation='relu',input_shape=(200,200,3)),
                                    tf.keras.layers.MaxPool2D(2,2),
                                    tf.keras.layers.Conv2D(32,(3,3),activation='relu'),
                                    tf.keras.layers.MaxPool2D(2,2),
                                    tf.keras.layers.Conv2D(64,(3,3),activation='relu'),
                                    tf.keras.layers.MaxPool2D(2,2),
                                    tf.keras.layers.Flatten(),
                                    tf.keras.layers.Dense(128,activation='relu'),
                                    tf.keras.layers.Dense(1,activation='sigmoid'),
                                    ])
```

Fig. 5. Neural network structure

IV. DATA PREPARATION

Before starting the network learning process, it is necessary to properly prepare the data that will be

given to the network input. To begin with, the collected data was divided into periods corresponding to the execution of a single work cycle by the robot (Fig. 4). So that the signal can be divided into cycles, the controller sends additional information about the beginning and end of the execution of the part of the program responsible for moving the object through the robot.

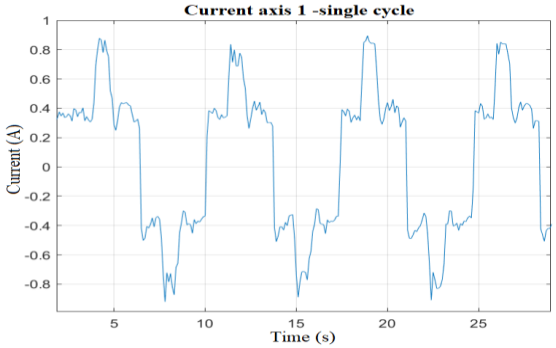


Fig. 6. Current during a single robot cycle

The divided signals are then transformed from the time domain to the frequency domain using a fast Fourier transform (Fig. 5). Finally, a single duty cycle is graphically represented in the form of a spectrogram and saved to a separate graphics file (Fig. 6). Each generated image was then given a label indicating the class to which the image belongs.

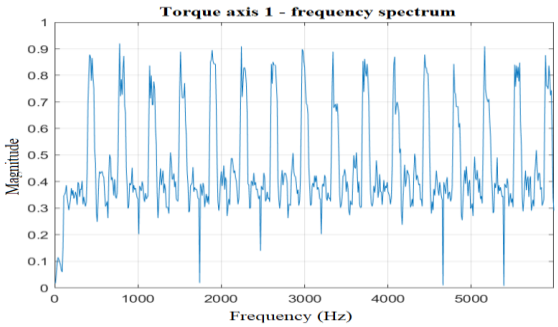


Fig. 7. Torque in the frequency domain

Initially, images and labels were saved in a single directory and then split into training, validation, and test sets. 66.6% of the images with labels were assigned to the training set, while 26.6% and 6.6% were assigned to the validation and test sets, respectively.

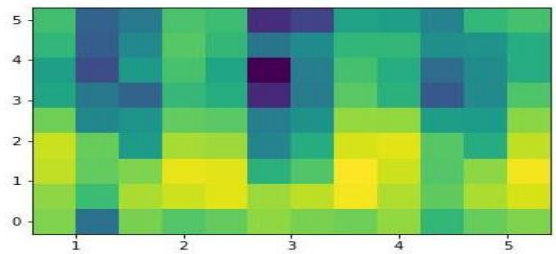


Fig.8. Spectrogram of the signal

V. RESULTS

A training data set was prepared, which included several hundred images of the correct robot cycle and cycles simulating an emergency situation. The network was prepared to perform a binary classification task. The learning process required classifying the given spectrograms into two classes: OK - correct duty cycle, NOK - duty cycle with failure. Once the spectrograms were assigned the correct classes, the new dataset was ready and learning could begin. The result of learning the network is influenced by many factors, the most important of which is the training data set. The larger the training set, the smaller the possibility that the network will overlearn and thus achieve better generality of performance. For this reason, as much data as possible is desired for learning. Fig. 7 shows the results of the best network learning approach.

Training on single GPU.
Initializing input data normalization.

Epoch	Iteration	Mini-batch Loss	Mini-batch Accuracy	Mini-batch RMSE	Validation Loss	Validation Accuracy
1	1	0.8745	14.94%	1.40	2.4430	99.52%
4	10	0.5759	19.82%	1.20		
7	20	0.3172	28.82%	0.89		
10	30	0.3289	49.81%	0.90		
14	40	0.1778	54.83%	0.61		
17	50	0.2091	69.82%	0.72	0.4894	99.86%
20	60	0.2327	75.85%	0.74		
24	70	0.0935	81.88%	0.44		
27	80	0.1698	89.15%	0.60		
30	90	0.1669	89.85%	0.62	0.3939	99.84%

Fig. 9. Learning progress

Accuracy (Mini-batch Accuracy) determines the percentage of training images that the network classified correctly, 90% is a satisfactory result. However, this is not an ideal result due to the large value of the loss (Mini-batch Loss), which should have a value closest to zero. This may mean that the training set should be increased. Below Fig. 8 and Fig. 9 show the result of classification of two images from the test base by the trained network.

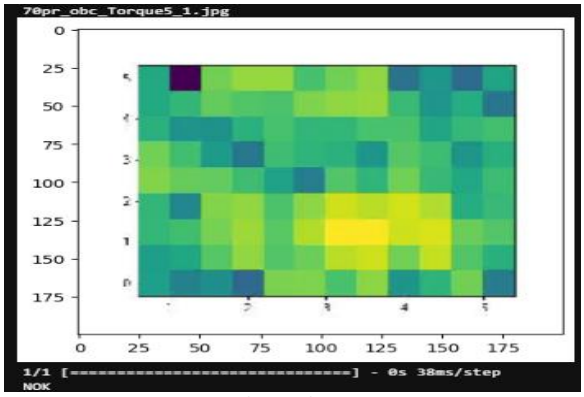


Fig. 10. Result of classification - NOK class

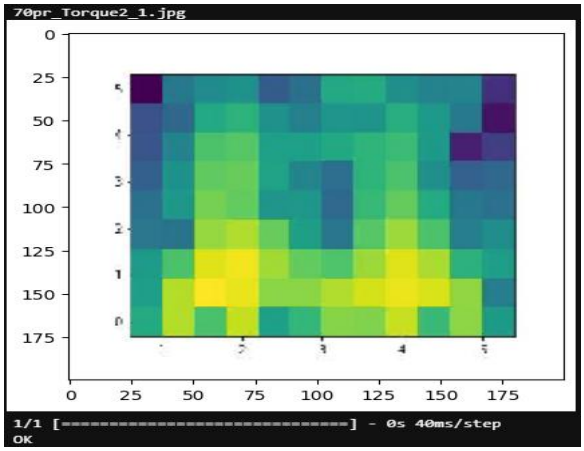


Fig. 11. Result of classification - OK class

The network was trained on different data sets, depending on the speed of the robot. The network performs better in classifying data collected for higher speeds. This is due to the fact that for higher speeds the values of currents and torques on the robot's drives reach larger amplitudes than at lower speeds, so the differences on the spectrograms between OK and NOK cycles are more noticeable.

VI. CONCLUSION

The accuracy rate of the implemented neural network algorithm, responsible for detecting abnormal cycles of the robot, is 90%, which is a satisfactory result. It is possible to increase the accuracy of the algorithm by expanding the training database. In addition, it is possible to modify the structure of the implemented network to even better solve the considered problem. The algorithm was tested for different robot speeds and returned the best results for high speeds. This is good news, since in industrial settings robots operate at the highest possible speed in order to achieve the greatest efficiency. Such a solution has a possibility of being implemented in the industry due to time and money savings. It can also improve safety by preventing the occurrence of failures early on. Further development of such a solution will potentially reduce the number of technical services to the required minimum.

A possible way forward could be to increase the amount of data that is analyzed by the network, such as signals from strain gauges that can be mounted on the robot's gripper. The algorithm will be further developed, and the results of the operation will be published in future works.

REFERENCES

[1] Mobley R K. (2002). Introduction to Predictive Maintenance. In Elsevier Science.

[2] Selcuk, S. (2017). Predictive maintenance, its implementation and latest trends. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 231(9). https://doi.org/10.1177/0954405415601640

[3] Fasuludeen Kunju, F. khan, Naveed, N., Anwar, M. N., &UlHaq, M. I. (2022). Production and maintenance in industries: impact of industry 4.0. Industrial Robot, 49(3). https://doi.org/10.1108/IR-09-2021-0211

[4] Hashemian, H. M., & Bean, W. C. (2011). State-of-the-art predictive maintenance techniques. IEEE Transactions on Instrumentation and Measurement, 60(10). https://doi.org/10.1109/TIM.2009.2036347

[5] Compare, M., Baraldi, P., & Zio, E. (2020). Challenges to IoT-Enabled Predictive Maintenance for Industry 4.0. IEEE Internet of Things Journal, 7(5). https://doi.org/10.1109/JIOT.2019.2957029

[6] Roblek, V., Meško, M., &Krapež, A. (2016). A Complex View of Industry 4.0. SAGE Open, 6(2). https://doi.org/10.1177/2158244016653987

[7] Çinar, Z. M., Nuhu, A. A., Zeeshan, Q., Korhan, O., Asmael, M., &Safaei, B. (2020). Machine learning in predictive maintenance towards sustainable smart manufacturing in industry 4.0. Sustainability (Switzerland), 12(19). https://doi.org/10.3390/su12198211

[8] Dalzochio, J., Kunst, R., Pignaton, E., Binotto, A., Sanyal, S., Favilla, J., & Barbosa, J. (2020). Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges. In Computers in Industry (Vol. 123). https://doi.org/10.1016/j.compind.2020.103298

[9] Drakaki, M., Karnavas, Y. L., Tziafettas, I. A., Linardos, V., &Tzionas, P. (2022). Machine Learning and Deep Learning Based Methods Toward Industry 4.0 Predictive Maintenance in Induction Motors: A State of the Art Survey. Journal of Industrial Engineering and Management, 15(1). https://doi.org/10.3926/jiem.3597

[10] Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., &Beghi, A. (2015). Machine learning for predictive maintenance: A multiple classifier approach. IEEE Transactions on Industrial Informatics, 11(3). https://doi.org/10.1109/TII.2014.2349359

[11] Lalik, K., &Wątorek, F. (2021). Predictive maintenance neural control algorithm for defect detection of the power plants rotating machines using augmented reality goggles. Energies, 14(22). https://doi.org/10.3390/en14227632

[12] Radanliev, P., de Roure, D., Nicolescu, R., Huth, M., & Santos, O. (2021). Artificial Intelligence and the Internet of Things in Industry 4.0. In CCF Transactions on Pervasive Computing and Interaction (Vol. 3, Issue 3). https://doi.org/10.1007/s42486-021-00057-3

[13] Munirathinam, S. (2020). Industry 4.0: Industrial Internet of Things (IIOT). In Advances in Computers (Vol. 117, Issue 1). https://doi.org/10.1016/bs.adcom.2019.10.010

[14] Lampropoulos, G., Siakas, K., & Anastasiadis, T. (2019). Internet of Things in the Context of Industry 4.0: An Overview. International Journal of Entrepreneurial Knowledge, 7(1). https://doi.org/10.2478/ijek-2019-0001

[15] Ghobakhloo, M. (2020). Industry 4.0, digitization, and opportunities for sustainability. In Journal of Cleaner Production (Vol. 252). https://doi.org/10.1016/j.jclepro.2019.119869

[16] Zhou, K., Liu, T., & Zhou, L. (2016). Industry 4.0: Towards future industrial opportunities and challenges. 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2015. https://doi.org/10.1109/FSKD.2015.7382284

[17] Büchi, G., Cugno, M., & Castagnoli, R. (2020). Smart factory performance and Industry 4.0. Technological Forecasting and Social Change, 150. https://doi.org/10.1016/j.techfore.2019.119790

[18] Milosevic, N. (2020). Introduction to Convolutional Neural Networks. In Introduction to Convolutional Neural Networks. https://doi.org/10.1007/978-1-4842-5648-0

[19] Chauhan, R., Ghanshala, K. K., & Joshi, R. C. (2018). Convolutional Neural Network (CNN) for Image Detection and Recognition. ICSCCC 2018 - 1st International Conference on Secure Cyber Computing and Communications. https://doi.org/10.1109/ICSCCC.2018.8703316

[20] Namatēvs, I. (2018). Deep Convolutional Neural Networks: Structure, Feature Extraction and Training. Information

Technology and Management Science, 20(1). https://doi.org/10.1515/itms-2017-0007

[21] Yi, Z. (2018). Evaluation and Implementation of Convolutional Neural Networks in Image Recognition. Journal of Physics: Conference Series, 1087(6). https://doi.org/10.1088/1742-6596/1087/6/062018

[22] Lalik, K., &Kozek, M. (2020). Cubic SVM neural classification algorithm for self-excited acoustical system. 15th International Conference Mechatronic Systems and Materials, MSM 2020. https://doi.org/10.1109/MSM49833.2020.9201724

[23] Lyu, S., & Liu, J. (2021). Convolutional recurrent neural networks for text classification. Journal of Database Management, 32(4). https://doi.org/10.4018/JDM.2021100105

[24] Momeny, M., Latif, A. M., Agha Sarram, M., Sheikhpour, R., & Zhang, Y. D. (2021). A noise robust convolutional neural network for image classification. Results in Engineering, 10. https://doi.org/10.1016/j.rineng.2021.100225

[25] Tian, Y. (2020). Artificial Intelligence Image Recognition Method Based on Convolutional Neural Network Algorithm. IEEE Access, 8. https://doi.org/10.1109/ACCESS.2020.3006097

[26] Sun, Y., Xue, B., Zhang, M., & Yen, G. G. (2020). Evolving Deep Convolutional Neural Networks for Image Classification. IEEE Transactions on Evolutionary Computation, 24(2). https://doi.org/10.1109/TEVC.2019.2916183

★★★