of_The_Industrial_Machine_Using_Support_Vector_Machine__SVM.pdf

by

FILE	OF_THE_INDUSTRIAL_MACHINE_USING_SUPPORT_VECTOR_MACHINE__SVM.PDF (859.38K)		
TIME SUBMITTED	25-DEC-2020 06:21PM (UTC+0700)	WORD COUNT	2977
SUBMISSION ID	1481210673	CHARACTER COUNT	15496

Fog Computing Concept Implementation in Work Error Detection System of The Industrial Machine Using Support Vector Machine (SVM)

Juprianus Rusman
Postgraduate Student of Electrical Engineering Department
Hasanuddin University
Makassar, Indonesia
rusmanj17d@student.unhas.ac.id

Zulkifli Tahir
Informatics Engineering Department
Hasanuddin University
Makassar, Indonesia
zulkifli@unhas.ac.id

A. Ejah Umraeni Salam
Electrical Engineering Department
Hasanuddin University
Makassar, Indonesia
ejahsalam@gmail.com

***Abstract*— The implementation of industry 4.0 to integrate various infrastructures is inseparable from the application of sensors, which is then followed by intensive data analysis. One form of the applications is the work error detection system for the industrial machine with the Fog Computing concept. In this study, the Fog Computing architecture was studied to support infrastructure integration for the development of smart industries. This study covers sensor network architecture, device communication, and smart computing with a robotic arm infrastructure. The sensor is installed to monitor the work process of the robot arm; then, the measurement values obtained sent to the fog device for analysis. The results of the analysis to determine the movement patterns then classified by the Support Vector Machine (SVM) method. The result of the system is that it can detect the movement of a robot arm. The test results of this study showed an average accuracy of 90%. Therefore this research can later be used as an initial demonstration and real learning to applied to other machine components and services in supporting the application of smart industries, especially in Indonesia, in the future.**

***Keywords*—work error detection, SVM, fog computing, smart industry, industry 4.0**

I. INTRODUCTION

Many machines are used to replace human operations in the industrial field. These machines were designed with the ability to do repetitive tasks[1]. The use of robot arms in production systems is aimed to increase productivity, quality, and safety in manufacturing[2]. Regardless of the application, there are significant implications for operator safety and productivity because the production system has the possibility of failure due to accidents or unexpected work errors[3]. The research conducted by Zulkifli found 14 machines experienced working errors from 16 total machines with the highest working error of one engine reaching nine times[4]. Erroneous work on the robot arm is mainly due to its inherent complexity. There are several sources of work errors in the robot arm, such as resources, mechanical, hydraulic, and aging machine life, which can cause a decrease in performance and unsafe operating conditions[5]. As a solution to early handling robotic arm work errors, a work error detection system is needed. Research related to the detection of robot arm working errors has been carried out by Takayuki Matsuno et al. analyzed the screw tightening errors in the robot manipulator by measuring the vibration of the bolt fastening torque using the SVM method [3]. Steffen W. Ruehl et al. Monitored robot service activities by estimating and identifying success or failure rates during the robot task execution process using the SVM and RBF methods [5]. Asha Vijayan et al., classify and predict robot articulator movements using the SVM and Naïve Bayes methods [6]. Ikbal E. et al. analyzed the noise and vibration of the robot joint to predict errors using the RBNN and SOMNN methods [7].

In the industrial revolution 4.0, robotic arm working error detection systems can utilize technology services that refer to the infrastructure for storing and processing data, namely Cloud Computing technology. Cloud Computing technology has been widely used for Big Data analysis. However, data stored in Cloud Computing has a challenge when the data is large in number and many different geographical locations, while the system requires fast feedback on the closest network to the sensor[8]. Low latency requires the system to detect work errors on time and bandwidth limits causing service interruption when many users request services from the data center at the same time[9]. In reality, the work error detection system must detect as soon as possible if there is a work error in the robot arm. Fog Computing technology proposed by Cisco, the development of the Cloud Computing paradigm exists to overcome these challenges. Fog Computing can work in applications with geographic location problems, ensuring performance and efficiency at the middle network-level ie, the network before reaching the main data processing site (data center)[10]. Dazhong Wu et al. have done the previous studies associated with the Fog Computing concept.

Monitoring in real-time high-performance scalability for diagnosis and prognosis of industrial machine health by utilizing wireless sensor networks, cloud computing, and machine learning [9]. Saad Bin Qaisar, et al. that proposed new architecture with a hybrid approach to learning models for network optimization using CNN and RNN methods[11], Stefan Mocanu, et al. that evaluated the ability of integration among devices in the industry. Fog node serves as a local cloud for real-time processing and controlling, and can be executed offline in a short time[12].

Based on the description in previous studies, then one of the industrial applications, the working error detection system of an industrial machine, is a relevant application that can take most of the benefits of the Fog Computing paradigm. This research proposes a Work Error Detection System of Industrial Machine with Fog Computing concept using Support Vector Machine (SVM) as well as a case study using a uArm Metal Robot.

II. MATERIALS AND METHODS

A. Fog Computing

In this study, we proposed the usage of Fog Computing concept. Node data collected in real-time for analytics and decision making. This concept is expected to be able to detect the occurrence of work errors from parameters that exceed the threshold. Detection of threshold values used the machine-learning algorithm embedded in the fog device. Fig. 1 shows the system architecture we used.

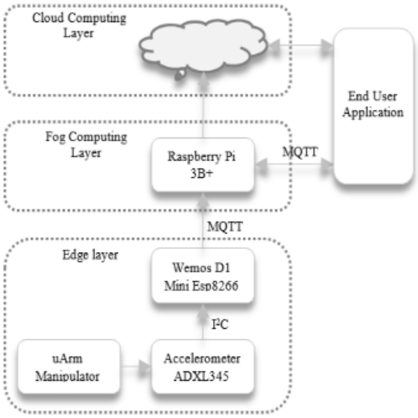


Fig. 1 Architecture system based on fog computing

- *Edge Layer*

  At this layer, a data acquisition device and a controller device installed. Micro Electromechanical Sensor (MEMS) ADXL345 accelerometer was mounted on the robot arm to monitor the movement of the robot arm. Wemos D1 Mini Esp8266 used as a data collection controller as well as a gateway to the fog device.

- *Fog Computing Layer*

  At this layer was provided a small computer device that functions to collect and analyze data and make decisions to notify if a working error of industrial machine is detected. Machine learning such as Neural Network, Support Vector Machine, K-Nearest Neighbors can be used to determine the initial pattern of machine work and identify working machine errors[10]. In this study, we used the Raspberry Pi 3B + device and the SVM algorithm.

- *Cloud Computing Layer*

  This layer is the central server, which, in principle, receives data from each lower layer device. Cloud Computing systems can be implemented using Hadoop, MapReduce, and so on. This service is expected to serve data on a broad scale with parallel system automation[13].

B. Accelerometer Data Acquisition

Data acquisition was carried out using the ADXL345 accelerometer sensor, which provides three-axis value outputs: X-axis, Y-axis, and Z-axis[14]. Measurement of the acceleration of robot motion, the accelerometer sensor is mounted on the end of the robot arm (end joint). The accelerometer sensor is connected to the Wemos D1 Mini Esp8266 (controller) using the Inter-Integrated Circuit (I^2C) protocol. When the robot arm was activated, the sensor measured the movement of the robot arm. Then the microcontroller sent data to the fog device. Fig. 2 shows the location of the sensor and microcontroller installation.

The values of sensor measurements were sent to the fog device (Raspberry Pi 3B+) using the Message Queue Telemetry Transport (MQTT) protocol. Mosquitto broker embedded in the fog device was used to receive data that recorded based on the average time of each movement of the robotic arm ie, 3800 ms, and then stored into the database.

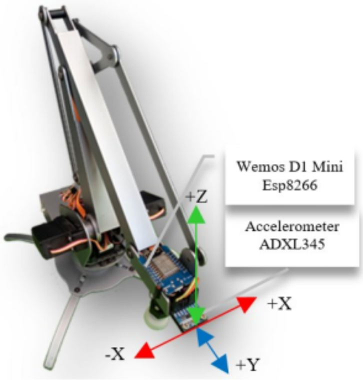
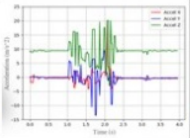
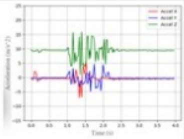
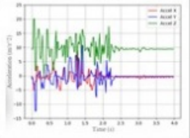
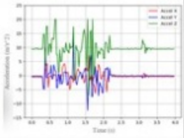
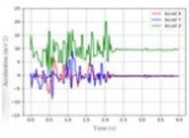
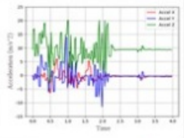
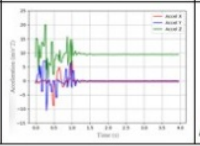
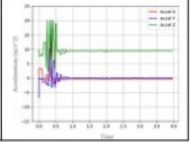


Fig. 2 Installation of the accelerometer sensor and microcontroller

Table 1 shows the condition of true (TM) and false (FM) movement, along with the measurement values of the robot movement. The first line shows the "start" movement, which is the movement from home to the source, the second line shows the "carry" movement, which is the movement from the source to the destination. The third line is the "take" movement that is a movement from the destination to the source, and in the fourth line is the "finish" movement that is a movement from the destination to home. Home is the initial position of the robot arm, the source is the position of taking the object, and the destination is the position to place the object.

TABLE I. ROBOT MOVEMENTS AND MEASUREMENT VALUES

Movement	Start Position	End Position			
		True movement	Values	False movement	Values
"Start"					
"Carry"					
"Take"					
"Finish"					

C. *Time Domain Features Extraction*

The time-domain feature considers data as a stationary signal to identify signal differences[15,16]. The time-domain feature was extracted from the raw data of the robot's movement to produce the dataset. The following features used.

- *Mean*

This feature takes the average value from the number of data samples. The average value is calculated by adding up the entire sample of data then divided by the number of samples. It can be written with equation (1).

$$mean(\mu) = \frac{1}{N}\sum_{n=1}^{N} Xn \qquad (1)$$

- *Variance*

Variance is a measure of how far data is scattered, can be written with equation (2).

$$var = \frac{1}{N-1}\sum_{n-1}^{N}(Xn-\mu)^2 \qquad (2)$$

- *Standard Deviation*

Standard deviation is the quality of measurement. The smaller standard deviation value is the better the quality, can be written with equation (3).

$$std(\sigma) = \sqrt{\frac{1}{N-1}\sum_{n-1}^{N}(Xn-\mu)^2} \qquad (3)$$

- *Root Mean Square* (RMS)

RMS is used to analyze measurement values. The steps to calculate the RMS value are to square each measurement value, calculate the average value, and finally to calculate the square root. It can be written by equation (4).

$$RMS = \sqrt{\frac{1}{N}\sum_{n=1}^{N} Xn^2} \qquad (4)$$

where *N*: the amount of data, *Xn*: the total amount of data.

After finishing the feature extraction process, then the next was to label the class for each movement. There were two class labels: class 1 label for TM, and class -1 label for FM. The labeled dataset used in the classification process. Table 2 shows the extraction results of robot movements (X-axis, Y-axis, and Z-axis) using the mean, variance, standard deviation, and root mean square features that have given a label of class.

TABLE II. DATASET

Index	Mean X	Mean Y	Mean Z	Variance X	Variance Y	Variance Z	Stdev X	Stdev Y	Stdev Z	RMS X	RMS Y	RMS Z	Class
1	-0.527	-0.040	9.099	1.858	5.728	8.346	1.363	2.393	2.889	0.726	0.201	3.017	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
120	-0.308	0.048	9.296	0.851	5.923	6.149	0.922	2.434	2.479	0.555	0.221	3.049	1
121	-1.233	-0.106	9.589	4.062	1.099	1.814	2.015	1.048	1.347	1.110	0.326	3.097	-1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
240	-0.599	-0.444	9.535	0.491	0.108	0.487	0.701	0.328	0.698	0.774	0.666	3.088	-1

D. Classification Using Support Vector Machine (SVM)

The next step was to classify the movement using the Support Vector Machine (SVM) algorithm. SVM algorithm is a classification problem-solving algorithm with the main principle of finding the best hyperplane that functions as a separator of two classes in the input space. The hyperplane can be a line in two dimensions and can be a flat plane in multiple planes[6]. There are four kernel functions in SVM, which are the linear kernel, Radial Basic Function (RBF), sigmoid, and polynomial. The purpose of classifying SVM is to conduct training using training data and make generalizations to predict class labels from test data. In the work error detection system of the industrial machine, there were two processes carried out namely the training and testing process. This process is shown in Fig. 3.

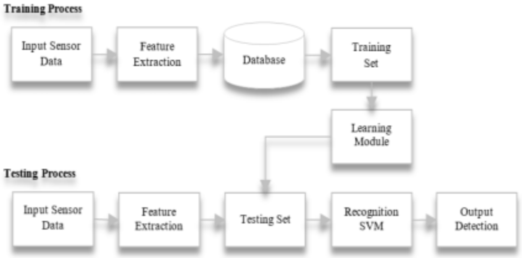


Fig. 3 Training and testing process on SVM

In this study, the RBF kernel was used with parameters C and gamma because it could produce better classification accuracy than other kernels[17]. Equation (5) is the equation used in the RBF kernel.

$$k(x_i x_j) = (-\gamma \, \|x - x'\|^2).\gamma \qquad (5)$$

where: x = training data feature and γ = gamma. The first step taken was to analyze the optimal value of parameter C, which is the margin distance, and γ, which is the acceleration of the function in the RBF kernel. For accurate results, a trial and error experiment was conducted to determine the optimum C and γ value. Optimum values obtained in this experiment: C = 1 and γ = 0.1. Fig. 4 shows a visualization of the classification of training data using RBF kernels with optimum C and γ.

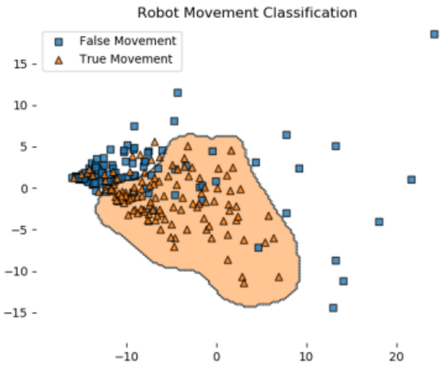


Fig. 4 Robot movement classification

The dataset consists of 240 data for the training process and 80 data for the test process. The testing process was carried out on four movements as follows: "start", "carry", "take", and "finish" with two classifications as TM, and FM. The test process can be seen in Table 3.

TABLE III. TEST ACCURACY

Movement	**Type of movement**	**Total**	**Detection**		
			TM	**FM**	**Accuracy (%)**
“Start”	TM	10	10	0	100
	FM	10	2	8	80
“Carry”	TM	10	10	0	100
	FM	10	2	8	80
“Take”	TM	10	10	0	100
	FM	10	2	8	80
“Finish”	TM	10	8	2	80
	FM	10	0	10	100
Average accuracy (%)					90

In table 3 shows the "start" movement of the TM category, ten movements were successfully detected as TM, which means the system's accuracy is 100%. Whereas in the FM category, the system detects 2 FM movements as TM (incorrect detection), and eight is successfully identified as FM (correct detection). This means that system accuracy is 80%. The same thing happens with the "carry" and "take" movements. Furthermore, in the "finish" movement of the TM category, eight movements were successfully detected as TM (correct detection) and detected two TM movements as FM (incorrect detection), which means the system accuracy was 80%. Whereas in the FM category, ten movements were successfully detected as FM, indicating the system accuracy was 100%. The result shows that the system could detect every movement of industrial machines both in TM and FM with an average accuracy of 90%.

III. CONCLUSION

Based on the test results in this study, which showed an average accuracy of 90%, it concluded that the development of an automated system for detecting the working error of the industrial machine is successfully carried out. Thus, the work error detection system using the Fog Computing concept with the SVM method used in the application of smart industries. In further research can be done by developing the results of this study in real-time for the monitoring process of industrial machinery while operating. The results of this study can also be used as an initial demonstration and real learning to be applied to other machine components and services in supporting the application of smart industries, especially in Indonesia, in the future.

REFERENCES

[1] Z. Lu, A. Chauhan, F. Silva, and L. S. Lopes, “A brief survey of commercial robotic arms for research on manipulation,” in *2012 IEEE Symposium on Robotics and Applications (ISRA)*, 2012, pp. 986–991.

[2] A. Jaber and R. Bicker, “Industrial Robot Fault Detection Based on Wavelet Transform and LabVIEW,” in *2014 1st International IEEE Conference on Systems Informatics, Modelling and Simulation*, 2014.

[3] T. Matsuno, J. Huang, and T. Fukuda, “Fault detection algorithm for external thread fastening by robotic manipulator using linear support vector machine classifier,” in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 3443–3450.

[4] Z. Tahir, *A hybrid maintenance management model in decision support system for small and medium food processing industries*. Universiti Teknikal Malaysia Melaka, 2010.

[5] S. W. Ruehl, Z. Xue, and R. Dillmann, “*Monitoring of manipulation activities for a service robot using supervised learning*,” in 2012 IEEE International Conference on Robotics and Automation, 2012, pp. 930–935.

[6] A. Vijayan, H. Singanamala, B. Nair, C. Medini, C. Nutakki, and S. Diwakar, “Classification of robotic arm movement using SVM and Naïve Bayes classifiers,” in *Third International Conference on Innovative Computing Technology (INTECH 2013)*, 2013, pp. 263–268.

[7] I. Eski, S. Erkaya, S. Savac, and S. Yildirim, “Fault detection on robot manipulators using artificial neural networks" - "Robotics and Computer-Integrated Manufacturing Volume 27, Issue 1, Pages 115-123,” Feb-2011.

[8] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog Computing and Its Role in the Internet of Things,” in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, New York, NY, USA, 2012, pp. 13–16.

[9] R. K. Naha *et al.*, “Fog Computing: Survey of Trends, Architectures, Requirements, and Research Directions,” *IEEE Access*, pp. 1–1, 2018.

[10] H. Atlam, R. Walters, and G. Wills, “Fog Computing and the Internet of Things: A Review,” *Big Data Cogn. Comput.*, vol. 2, no. 2, p. 10, Apr. 2018.

[11] S. B. Qaisar and M. Usman, “Fog Networking for Machine Health Prognosis: A Deep Learning Perspective,” in *Computational Science and Its Applications – ICCSA 2017*, 2017, pp. 212–219.

[12] S. Mocanu, G. Geampalia, O. Chenaru, and R. Dobrescu, “Fog-Based Solution for Real-Time Monitoring and Data Processing in Manufacturing,” in *2018 22nd International Conference on System Theory, Control and Computing (ICSTCC)*, Sinaia, 2018, pp. 504–509.

[13] M. Lavassani *et al.*, “Combining Fog Computing with Sensor Mote Machine Learning for Industrial IoT,” *Sensors*, vol. 18, no. 5, p. 1532, May 2018.

[14] “ADXL345.pdf.” Available: https://www.sparkfun.com/datasheets/Sensors/Accelerometer/ADXL345.pdf.

[15] C. Altin and O. Er, “Comparison of Time and Frequency Domain Feature Extraction Methods on Elbow Gesture’s EMG,” *Eur. J. Interdiscip. Stud.*, vol. 5, p. 35, Aug. 2016.

[16] W. Caesarendra and T. Tjahjowidodo, “A Review of Feature Extraction Methods in Vibration-Based Condition Monitoring and Its Application for Degradation Trend Estimation of Low-Speed Slew Bearing,” *Machines*, vol. 5, no. 4, p. 21, Dec. 2017.

[17] R. Gholami and N. Fakhari, “Chapter 27 - Support Vector Machine: Principles, Parameters, and Applications,” in *Handbook of Neural Computation*, P. Samui, S. Sekhar, and V. E. Balas, Eds. Academic Press, 2017, pp. 515–535.

of_The_Industrial_Machine_Using_Support_Vector_Machine__S.

ORIGINALITY REPORT

%7	%	%7	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	"Proceedings of the Future Technologies Conference (FTC) 2020, Volume 2", Springer Science and Business Media LLC, 2021 Publication	%1
2	"ISRITI 2019 Paper Titles", 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), 2019 Publication	%1
3	Amin B. Abkenar, Seng W. Loke, Arkady Zaslavsky, Wenny Rahayu. "GroupSense", ACM Transactions on Embedded Computing Systems, 2019 Publication	%1
4	Alaa Abdulhady Jaber. "Design of an Intelligent Embedded System for Condition Monitoring of an Industrial Robot", Springer Science and Business Media LLC, 2017 Publication	%1
5	Stefan Mocanu, Giorgiana Geampalia, Oana	%1

Chenaru, Radu Dobrescu. "Fog-Based Solution for Real-Time Monitoring and Data Processing in Manufacturing", 2018 22nd International Conference on System Theory, Control and Computing (ICSTCC), 2018

Publication

---

6 Vinsensius Letsoin, Faizal Arya Samman, A. Ejah Umraeni Salam. "Three-Phase DC-AC Inverter with Low Power Dissipation Filter for Photovoltaic-Based Micro-Grid Scale Electric Power System", 2018 Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS), 2018 %1

Publication

---

7 Khoa Nguyen, Dat Tran, Wanli Ma, Dharmendra Sharma. "The impact of data fragment sizes on file type recognition", 2014 10th International Conference on Natural Computation (ICNC), 2014 %1

Publication

---

8 "Intelligent and Cloud Computing", Springer Science and Business Media LLC, 2021 <%1

Publication

---

9 Zulkifli Tahir, Amil Ahmad Ilham, Muhammad Niswar, Dewiani Djamaluddin. "The Perspectives of Design Thinking Method on Computer Network Design and Implementation", <%1

2019 Fourth International Conference on Informatics and Computing (ICIC), 2019
Publication

EXCLUDE QUOTES	ON	EXCLUDE MATCHES	< 5 WORDS
EXCLUDE BIBLIOGRAPHY	ON		