IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.17, No.4, October 2023, pp. 337~346
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: https://doi.org/10.22146/ijccs.82697

Classification Methods Performance On Logistic Package State Recognition

Muhammad Auzan*[1], Dzikri Rahadian Fudholi[2], Paulus Josianlie P[3], M Ridho Fuadin[4]
[1,2]Department of Computer Science and Electronics, FMIPA UGM, Yogyakarta, Indonesia
[3,4]Bachelor Program of Electronics and Instrumentation, FMIPA UGM, Yogyakarta, Indonesia
e-mail: ***[1]muhammadauzan@ugm.ac.id**, [2]dzikri.r.f@ugm.ac.id,
[3]paulusjosianlie@mail.ugm.ac.id, [4]fuadin.ridho@gmail.id

Abstrak

Di sektor distribusi, kegiatan pengalaman paket logistik, seperti transportasi, distribusi, penyimpanan, pengemasan, dan penanganan. Meskipun proses tersebut memiliki prosedur operasional yang masuk akal, terkadang pengalaman paket dapat diabaikan. Pengabaian tersebut sulit diidentifikasi karena banyak paket yang berjalan secara bersamaan, dan tidak semua proses dipantau. Sebuah Inertia Measurement Unit dipasang di dalam paket untuk mengumpulkan tiga data percepatan dan rotasi. Data kemudian diberi label secara manual ke dalam empat kelas: penanganan yang benar, jatuh vertikal, dan lemparan dan jatuh berputar. Kemudian, menggunakan validasi silang, sepuluh klasifikasi digunakan untuk menghasilkan model untuk mengklasifikasikan status paket logistik dan mengevaluasi skor akurasi. Data yang dikumpulkan menunjukkan bahwa hanya data akselerometer yang dapat membedakan antara status paket. Namun, sulit untuk membedakan antara jatuh bebas dan lemparan. Klasifikasi hanya menggunakan data akselerometer untuk meminimalkan waktu proses. Klasifikasi penanganan yang benar memberikan hasil yang baik karena pola data memiliki sedikit variasi. Namun, data lemparan, jatuh bebas, dan berputar memberikan hasil yang lebih rendah karena pola mirip satu sama lain. Akurasi rata-rata dari sepuluh klasifikasi adalah 78,15, dengan deviasi rata-rata 4,31. Klasifikasi terbaik untuk penelitian ini adalah Gaussian Process, dengan akurasi rata-rata 94,4% dan deviasi 3,5%.

Kata kunci*— klasifikasi, logistik, Internet of things, kesalahan penanganan*

Abstract

In the distribution sector, logistic package experience activities, such as transport, distribution, storage, packaging, and handling. Even though those processes have reasonable operational procedures, sometimes the package experience mishandling. The mishandling is hard to identify because many packages run simultaneously, and not all processes are monitored. An Inertial Measurement Unit (IMU) is installed inside a package to collect three acceleration and rotation data. The data is then labeled manually into four classes: correct handling, vertical fall, and thrown and rotating fall. Then, using cross-validation, ten classifiers were used to generate a model to classify the logistic package status and evaluate the accuracy score. It is hard to differentiate between free-fall and thrown. The classification only uses the accelerometer data to minimize the running time. The correct handling classification gives a good result because the data pattern has few variations. However, the thrown, free-fall and rotating data give a lower result because the pattern resembles each other. The average accuracy of the ten classifications is 78.15, with a mean deviation of 4.31. The best classifier for this research is the Gaussian Process, with a mean accuracy of 94.4 % and a deviation of 3.5 %.

Keywords— *Classification, Logistic IoT, IMU, Mishandling.*

Received February 28th,2023; Revised October 31th, 2023; Accepted October 31th, 2023

1. INTRODUCTION

The rises of e-commerce makes nearly all transactions from anywhere [1], [2]. Supported by the government that wants to decrease poverty by evenly distributing goods, the number of goods transported continuously increases yearly. Even though goods transportation always has an excellent standard operation procedure, sometimes there is a mishandling in the process [3], [4]. However, because many goods are distributed every hour, monitoring is ineffective if done using visuals or human resources [5].

One effective way to monitor many objects that can be done from everywhere and at any time is the Internet of Things (IoT) [6]. Using sensors and networks, IoT can send the data to a server and real-time monitor by an operator [5]. The manager can also use the data to plan future business. The customer can know what happened to their package. IoT in logistic package transportation will improve reliability and transport resilience when encountering a problem [7].

Mishandling in transportation is a motion from an external force that makes the object experience a dangerous motion, such as free fall, throwing, and rotating. The mishandling can harm the package by making the object break, leak or cause other damage. Because it is sourced from an external force, a sensor that can sense a motion in an object is the Inertial Measurement Unit (IMU) [8]–[10].

Detection and classification using IMU are done by analyzing the acceleration in the x, y, and z axes. Measurement data is collected for some time. Then, the sequence, value, and direction are analyzed using many methods, such as Machine Learning, Deep Learning, and Digital Signal Processing [10]. The result was also affected by the sensor resolution and sampling rate [11]. The higher resolution will make the detection not sensitive enough to detect a minor mishandling. The higher sampling rate will make the data more extensive, and much data not be needed in the process. Therefore, data for mishandling classification need to consider the specification of the sensor.

With the same case but a different object, Human Activity Recognition (HAR) research has been done using the image. The accuracy result is between 77% and 82% [12]. Other than that, processing using image takes too much time, counted until one minute. Suppose it is implemented in a simple logistic process; it will make the process slower.

Using deep learning methods to do activity recognition has its advantage. Using raw data as input and the process done by deep learning can also yield a fast process[10], [12], [13]. Otherwise, deep learning needs more data than machine learning. The number of data can increase by using augmentation [14].

Whether deep learning or machine learning, it is proven that human activity can be recognized. Although different, hypothetically, logistic package state recognition can also be classified using machine learning or deep learning. However, the challenge for logistic package detection is not only the classification success but also the computation and the lack of data because it is implemented in a small device. Therefore this research focuses on building a model to classify logistic package states using IMU using ten different machine learning classifiers.

2. METHODS

2.1 Package Design

The logistic package comes in many kinds of shapes and sizes. In this research, the size and shape are not constrained. As such, the size and shape of the package are different in each data collection. The inside of the logistic package is randomized with a heterogeneous object. The IoT device to collect the data is placed on the bottom side of the package, as shown in Figure 1.

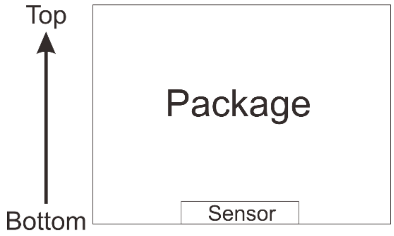

Figure 1 Inertia measurement unit placement inside the package.

This research uses the small and low power Inertia Measurement Unit (IMU) from Bosch BMI270 built-in Metamotion S+ device. Metamotion S+ is 36mm x 27 mm x 10 mm. Thus the size will not hinder the package content.

2.2 Inertia measurement unit specification

BMI270 has a 16-bit resolution for each accelerometer and gyro scope with many sensitivities. The BMI sensitivity is programmed to get data with LSB $6.10 \, x \, 10^{-5}$ g for the accelerometer $7.63 \, x \, 10^{-3}$. The data is collected by a microcontroller in the Metamotion S+ device and then sent to the smartphone in the CSV file format. All configuration used in the data collection phase is shown in Table 1. The setting on the smartphone using the Metabase app is shown in Figure 2.

Table 1 Data specification

No	Specification	Details
1	Data type	3-Axis Accelerometer 3-Axis Gyroscope
2	Frequency Sampling	25.000 Hz
3	Accelerometer Sensitivity	4 g
4	Gyroscope Sensitivity	500 °/s

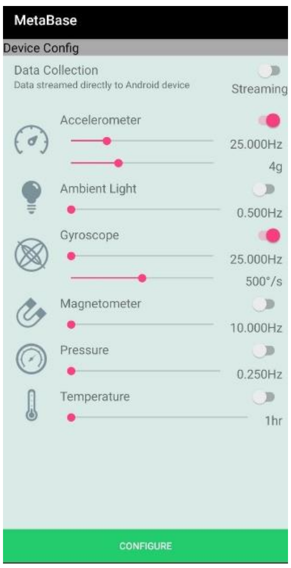

Figure 2 Metabase Configuration

2.3 Data Collection

Four different motions are recorded using Metamotion S+: steady, vertical fall, throw, and rotating fall. The motion data is achieved by following some sequence. The first sequence is for the person that handles the package to stand with any motion except the mishandling motion for the first three seconds. This motion is considered steady because it is categorized as safe for the package, even though it is carried in unique ways. The second sequence is the mishandling motion after the third second. The last sequence is for the package to be left alone until 10 seconds pass, counted from the beginning of the data-collecting process. Figure 3 – Figure 5 describes the motion of each mishandling scenario.

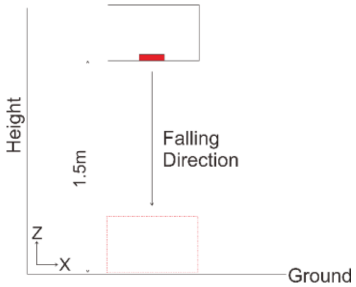

Figure 3 Vertical Fall Motion Scenario

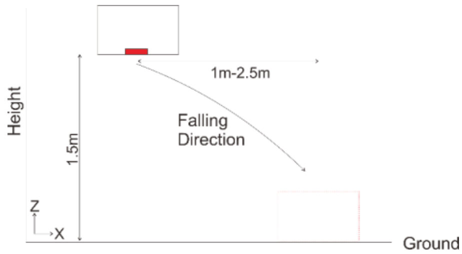

Figure 4 Throw Motion Scenario

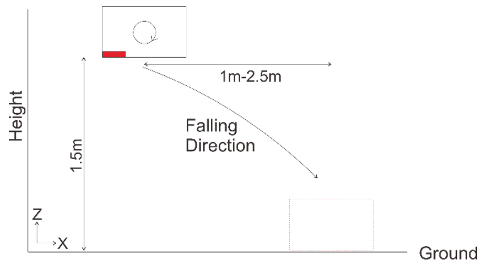

Figure 5 Rotation Motion Scenario

The data result will have three phases for only a one-axis accelerometer, as shown in Figure 6. The first phase is stable when carried by a person. Meanwhile, the third phase is stable when placed on the floor.

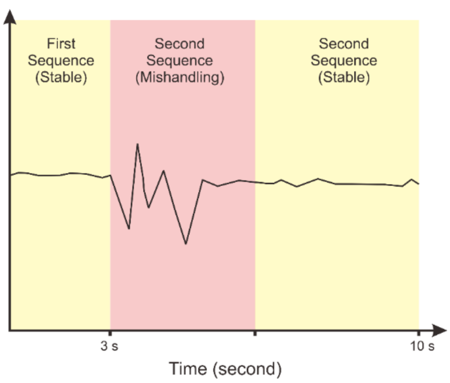

Figure 6 One axis data plot illustration

2.4 Data Augmentation

There is only one mishandling motion for each data collection, and each sequential data is divided into three. The three data are steady before mishandling when the mishandling happened and steady state after the mishandling. There are 20 data for each mishandling motion, with 60 data. Each data is split into three part and become 180 data.

After the data is collected, the next phase is to validate the data. Some data shows that there is an error in data value. The error data value is then discarded from the dataset. Another aspect of being counted is the unbalance data between the steady and mishandling. Unbalance data will affect the result training process. As such, the collected motion data is discarded because it exceeds the average of mishandled data.

The total data after data validation is too tiny for the machine learning training process. The data augmentation method was used in this research to increase the amount of data. The augmentation process uses Tsaug in python programming to add noise and signal dropout. Adding noise will change the overall signal by adding random noise with some constraints. Dropout will add a zero value in some signal parts to simulate the miss data collecting during real-time implementation. The configuration of noise and dropout can be seen in Table 2.

Table 2 Data Augmentation Configuration

No	Specification	Details	Value
1	Noise	Lower Limit	1%
		Higher Limit	5%
2	Dropout	Probability	0.1
		Value	0
		Size	2 – 5 Data

2.5 Data Classification

Classification can be done using various kinds of methods. This research will classify the logistic package state with several classifiers to see the overall performance using the proposed data. Ten classifiers were used in this research. Each configuration was variated to see the overall response when using the proposed data. Each classifier used in this research is explained in Table 3.

Table 3 Data Classifier

No	Classifier
1	Decision Tree
2	KNN
3	Naïve Bayes
4	Random Forest
5	Support Vector Machine
6	Logistic Regression
7	Multi-Layer Perceptron
8	Gaussian Process
9	Ada Boost
10	Quadratic Discriminant

2.5 Cross Validation

Cross-validation is a statistical method for validating the model with different train and test datasets. The cross-validation result can show us the robustness of the classifier. This research uses 5-fold cross-validation that will divide the entire dataset into five same-size subsets. Each process will use four subsets as the training data and one subset as the testing. There will be five processes, each with a different training and testing subset. The illustration of the 5-fold cross-validation used in this research is shown in Figure 7.

Dataset				
1	2	3	4	5
Train				Test
Train			Test	Train
Train		Test	Train	
Train	Test	Train		
Test	Train			

Figure 7 One axis data plot illustration

3. RESULTS AND DISCUSSION

This section provides the result of this research and explains in detail every result served in a table or graph format. The first thing explained in this section is the data collection process and the data shape. The second is the performance of ten classifiers used in this research.

This research uses an inertia measurement unit to collect three kinds of data: vertical fall, rotating fall, and throw motion data. The data collected is from 3 Axis accelerometers and three-axis gyro scope. Based on the data we get, correct package handling does not have a sudden change in any axis. However, the data can have a high acceleration value, as shown in Figure 8. The correct handling happens from time 0 until 60 and from 90 until 120. The acceleration at that time is not 0 and still has some fluctuation. From Figure 8, correct handling does not affect the gravitation value's significance but the pattern over time.

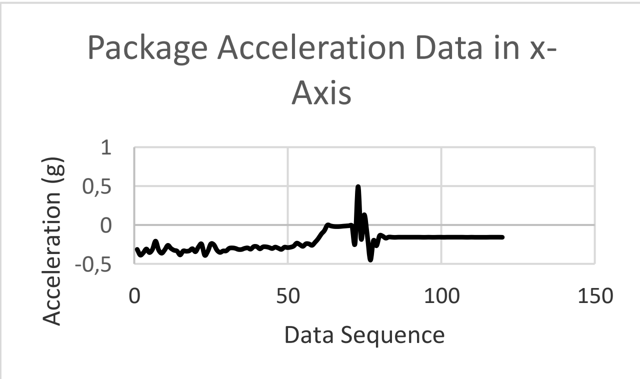

Figure 8 Ten-second raw data from the accelerometer in x-Axis

The data from correct handling can easily be recognized because of how the sensor data behave. Otherwise, differentiating between mishandling motions is more challenging. The data shows only a slight difference between throw and vertical fall. The value between throw and vertical fall is different in the data sequence. The vertical fall acceleration will experience a sudden stop at the end after constant acceleration.

The throw acceleration will experience deceleration before experiencing another acceleration in the opposite direction, as shown in Figure 9 In the first throw phase, the package experience slight changes in the acceleration, as shown in Figure 9, from data 60 until 65, when the acceleration is going up. After the package reaches the peak velocity, it will go down until it lands on the ground, as shown by the data from data number 80 until 100.

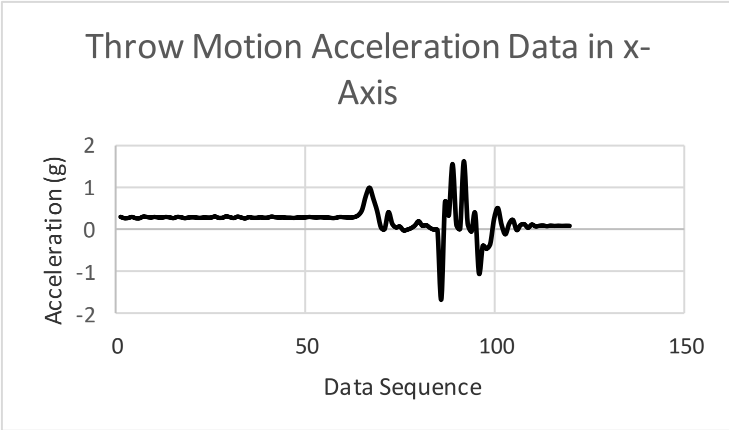

Figure 9 Throw data in x-Axis for 10 second

As for rotating data, the acceleration will happen in all axis alternately, as shown in Figure 10, Figure 11, and Figure 12. Rotating motion has its unique pattern. The data from the axis where the rotation happens is more stable than the other two. It can be shown in Figure 10 in data 65 until 80, where the oscillation is small compared to Figure 11 and Figure 12.

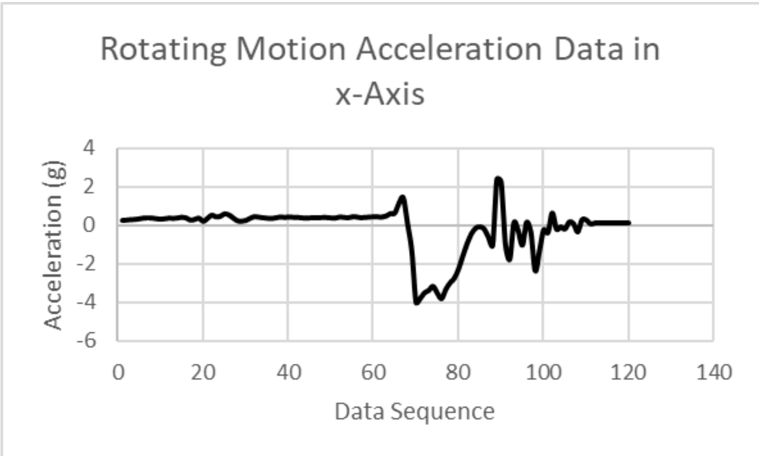

Figure 10 Rotating data in x-Axis for 10 second

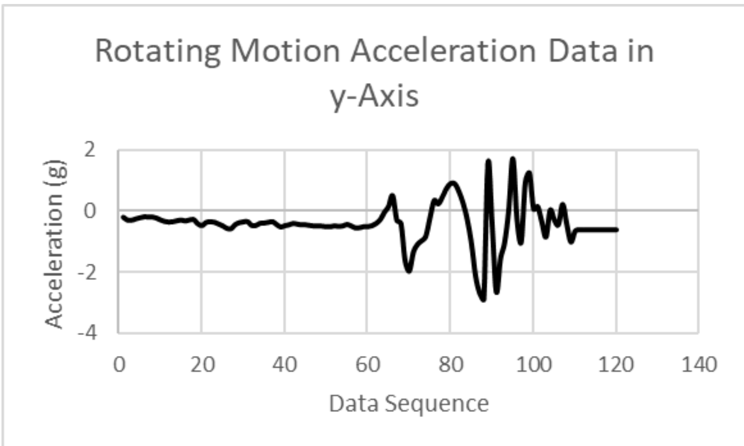

Figure 11 Rotating data in x-Axis for 10 second

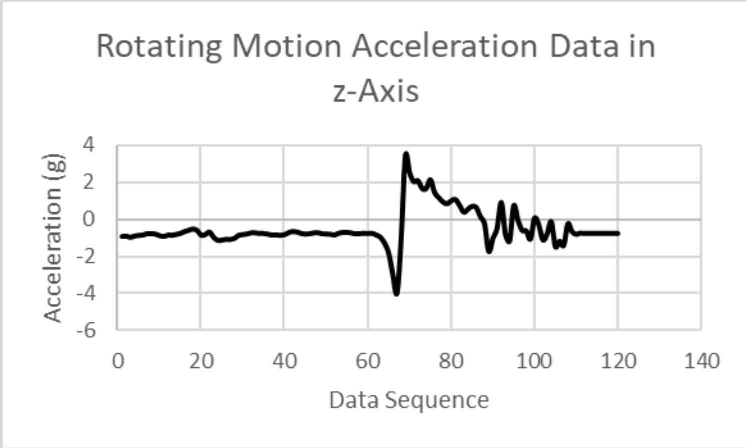

Figure 12 Rotating data in x-Axis for 10 second

Those analyses become the basis of data slicing in the following process. The data is divided into correct handling and mishandling. From 120 long data, it will be divided into two parts, each having 60 long data labeled as shown in Figure 13.

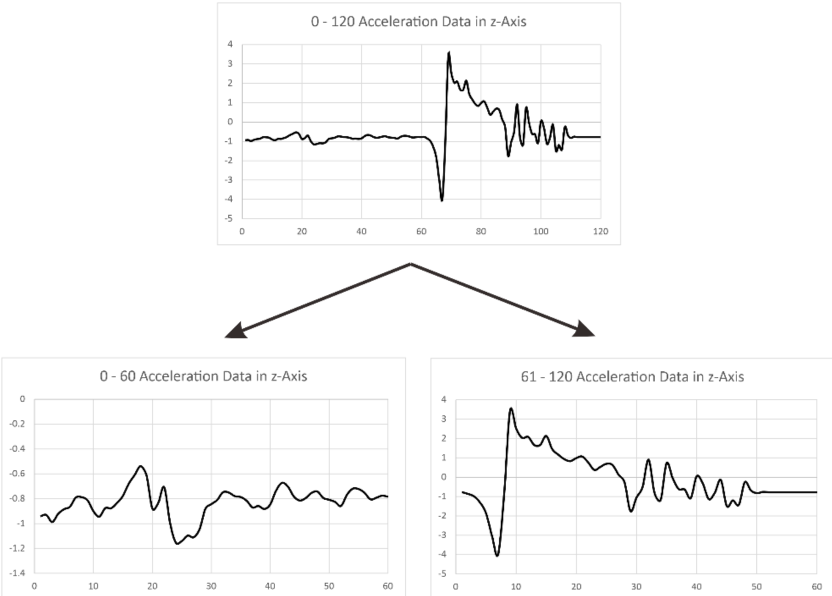

Figure 13 Sensor data split into two phase

The process then goes to augmentation to add noise and dropout. Dropout simulates when the sensor has an error reading, so the result is zero. It also becomes a new challenge for the classifier to classify when there is a data loss in a real-time application. Figure 14 shows the difference between raw data and augmented data.

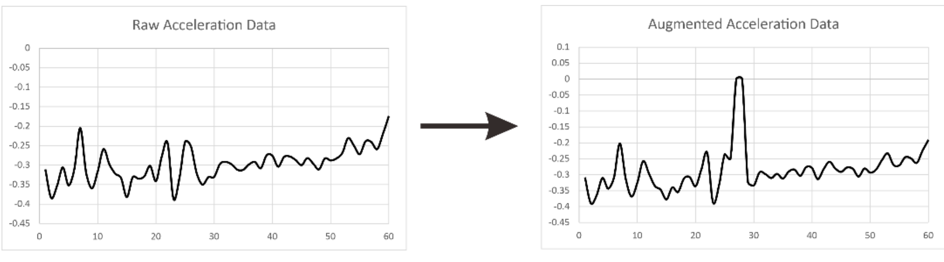

Figure 14 x-Axis Accelerometer Raw Data Augmentation

Table 4 Data Classifier

No	Classifier	Average Accuracy
1.	KNN	70.9% (+/- 5.8)
2.	Decision Tree	81.7 % (+/- 4.9) %
3.	Naïve Bayes	73.7 % (+/- 4.7) %
4.	Random Forest	83.6 % (+/- 4.3) %
5.	Support Vector Machine	87.4 % (+/- 4.2) %
6.	Multi-Layer Perceptron	85.9 % (+/- 2.0) %
7.	Logistic Regression	81.3 % (+/- 6.3) %
8.	Gaussian Process	94.4 % (+/- 3.5) %
9.	Ada Boost	59.2 % (+/- 4.2) %
10.	Quadratic Discriminant	63.4 % (+/- 3.2) %

The result from each classifier, the gaussian process classifier, gets the best result with 94.4 % (+/- 3.5) %. The gaussian process is a non-parametric classifier except for selecting the kernel. There are three kernels in the gaussian process: RBF, Rational Quadratic, and Matern. The gaussian process implements Gaussian probability and Bayesian to update the belief. This result is the strength of the Gaussian Process, where the gaussian distribution can also model the uncertainty that other classifiers do not have.

4. CONCLUSIONS

In this work, we evaluate ten machine-learning methods to classify the state of the logistic package using an inertial measurement unit sensor. The process consists of collecting three mishandling data: vertical fall, rotation fall, and throw. The data then goes to the augmentation process to add noise and dropout signal. The result is 708 data divided into four classes. The data was then evaluated using KNN, Decision Tree, Naïve Bayes, Random Forest, Support Vector Machine, Multi-Layer Perceptron, Logistic Regression, Gaussian Process, Ada Boost and Quadratic Discriminant. The performance of each classifier was evaluated using an accuracy score using 5 Fold Cross Validation. The result shows that Gaussian Process gives the best result with 94.4 % (+/- 3.5) % accuracy. The Gaussian Process uses RBF as the kernel, which is good enough to classify sequence data from 3-axis accelerometer data.

Motion or activity recognition is a challenging topic to research because there are many different kinds of motion but with slight differences in data. Especially when the goal is to be implemented in a small device with limited computation ability. Using short data will produce less computation, but classification results have lower accuracy and vice versa. The data from the sensor is also another challenging problem. External force other than gravitation will produce a greater force and make the sensor sensitivity analysis difficult because higher force needs a more sensitive sensor, but a more sensitive sensor cannot detect small force that also affects the logistic package.

ACKNOWLEDGEMENTS

The authors would like to thank the UGM Fakultas Matematika dan Ilmu Pengetahuan Alam for providing "financial support" to this research through funding assistance from legal entities of Hibah Penelitian Tahun 2022.

REFERENCES

[1] M. Lafkihi, S. Pan, dan E. Ballot, “Freight transportation service procurement: A literature review and future research opportunities in omnichannel E-commerce,”

Transp. Res. Part E Logist. Transp. Rev., vol. 125, hal. 348–365, Mei 2019, doi: 10.1016/j.tre.2019.03.021.

[2] R. Bandara, M. Fernando, dan S. Akter, "Privacy concerns in E-commerce: A taxonomy and a future research agenda," *Electron. Mark.*, vol. 30, no. 3, hal. 629–647, Sep 2020, doi: 10.1007/s12525-019-00375-6.

[3] A. Panghal, S. Manoram, R. S. Mor, dan P. Vern, "Adoption challenges of blockchain technology for reverse logistics in the food processing industry," *Supply Chain Forum An Int. J.*, hal. 1–10, Jun 2022, doi: 10.1080/16258312.2022.2090852.

[4] W. Prosser *et al.*, "Redesigning immunization supply chains: Results from three country analyses," *Vaccine*, vol. 39, no. 16, hal. 2246–2254, Apr 2021, doi: 10.1016/j.vaccine.2021.03.037.

[5] D. S.-M. Machado, D. Moreira, H. Castro, dan A. Barao, "Real-Time Logistic Monitoring (RTLM)," in *2020 15th Iberian Conference on Information Systems and Technologies (CISTI)*, Jun 2020, hal. 1–3, doi: 10.23919/CISTI49556.2020.9140441.

[6] H. Manoharan, Y. Teekaraman, I. Kirpichnikova, R. Kuppusamy, S. Nikolovski, dan H. R. Baghaee, "Smart Grid Monitoring by Wireless Sensors Using Binary Logistic Regression," *Energies*, vol. 13, no. 15, hal. 3974, Agu 2020, doi: 10.3390/en13153974.

[7] D. S.-M. Machado, D. Moreira, H. Castro, dan A. Barão, "Real-Time Logistic Monitoring: System Prototype Overview," *J. Inf. Syst. Eng. Manag.*, vol. 6, no. 2, hal. em0139, 2021, doi: 10.29333/jisem/9671.

[8] J.-W. Cui, Z.-G. Li, H. Du, B.-Y. Yan, dan P.-D. Lu, "Recognition of Upper Limb Action Intention Based on IMU," *Sensors*, vol. 22, no. 5, hal. 1954, Mar 2022, doi: 10.3390/s22051954.

[9] C. Mummadi *et al.*, "Real-Time and Embedded Detection of Hand Gestures with an IMU-Based Glove," *Informatics*, vol. 5, no. 2, hal. 28, Jun 2018, doi: 10.3390/informatics5020028.

[10] T. T. Alemayoh, M. Shintani, J. H. Lee, dan S. Okamoto, "Deep-Learning-Based Character Recognition from Handwriting Motion Data Captured Using IMU and Force Sensors," *Sensors*, vol. 22, no. 20, hal. 7840, Okt 2022, doi: 10.3390/s22207840.

[11] E. D'Amato, V. A. Nardi, I. Notaro, dan V. Scordamaglia, "A Particle Filtering Approach for Fault Detection and Isolation of UAV IMU Sensors: Design, Implementation and Sensitivity Analysis," *Sensors*, vol. 21, no. 9, hal. 3066, Apr 2021, doi: 10.3390/s21093066.

[12] I. A. Putra, O. D. Nurhayati, dan D. Eridani, "Human Action Recognition (HAR) Classification Using MediaPipe and Long Short-Term Memory (LSTM)," vol. 43, no. 2, hal. 190–201, 2022, doi: 10.14710/teknik.v43i2.46439.

[13] N. Gupta, S. K. Gupta, R. K. Pathak, V. Jain, P. Rashidi, dan J. S. Suri, *Human activity recognition in artificial intelligence framework: a narrative review*, vol. 55, no. 6. Springer Netherlands, 2022.

[14] O. Steven Eyobu dan D. Han, "Feature Representation and Data Augmentation for Human Activity Classification Based on Wearable IMU Sensor Data Using a Deep LSTM Neural Network," *Sensors*, vol. 18, no. 9, hal. 2892, Agu 2018, doi: 10.3390/s18092892.