

Estimation of Axis Roll Pitch of GY-91 IMU Sensor Reading Using Kalman Filter

Mochamad Mobed Bachtiar^{1*}, Iwan Kurnianto Wibowo², Yusuf Rifa'i³, Daniswara Prasetya Subagja⁴, Nanda Alfi Syahriyah⁵
^{1,2,3,4,5}Department of Informatics and Computer Engineering, Politeknik Elektronika Negeri Surabaya, Indonesia

¹mobed@pens.ac.id*; ²eone@pens.ac.id; ³yusufirifai43@gmail.com; ⁴daniswpr@gmail.com; ⁵nandaafliis@gmail.com

*corresponding author

ABSTRACT

The Inertial Measurement Unit (IMU) sensor is a tool used to measure the speed and acceleration of an object in 3 dimensions (x, y, z). IMU sensors are often used in robotics, drone control, autonomous vehicles, and augmented reality applications. Usually, the data obtained from the IMU sensor is contaminated by interference and noise, which can reduce measurement accuracy. Kalman Filter is a statistical method used to combine measurement data with a mathematical system model to produce better estimates. In the IMU context, the Kalman Filter removed interference and noise affecting acceleration and speed data so that IMU sensor data could be estimated more accurately. This algorithm predicts the next data state based on previous data and updates the prediction with new measurement data. The measurement implementation in this research is the IMU sensor on the GY-91 module to determine the object's tilt on the pitch, roll, and yaw axes during flight. The ARM STM32F407VGT6 microcontroller pin reads the sensor, and then the estimation and prediction process is carried out using the Kalman filter algorithm. With the parameters Kalman Measurement Error = 1, Estimation Error = 0.12, and Covariance Process = 0.4, it can predict the reading results from the IMU sensor well.

Keywords : Sensor, IMU, Kalman Filter, Estimation, Prediction, ARM.

This is an open-access article under the [CC-BY-SA](#) license.



Article History

Received : Oct, 21st 2023

Accepted : Nov, 28th 2023

Published : Nov, 30th 2023

I. INTRODUCTION

Inertial Measurement Unit (IMU) sensors are an essential component in various applications involving movement and orientation monitoring, their implementation is mostly for unmanned aircraft [1][5][12][13], augmented reality[2][3][11][14], robotics [4][7][8][9], and autonomous vehicles [6]. However, the data provided by IMU sensors tends to be polluted by interference and noise, resulting in less accurate estimated values. In this context, the Kalman Filter algorithm is an effective solution to improve the quality of IMU sensor data and make more reliable predictions [10]. Kalman Filter is a recursive statistical algorithm used to combine measurement data with a mathematical model of the system to produce better estimates. It works by considering two main stages: prediction and update. In the prediction stage, the algorithm estimates the next state based on the previous state and the system model. Then, in the update stage, new measurement data is used to update those predictions, giving greater weight to more accurate data. The application of the Kalman Filter to IMU sensor data [10] is to estimate and predict data values that were previously inaccurate due to noise into data that is accurate and ready to be processed by the main system. This filter helps measure acceleration, angular velocity, and object orientation with higher accuracy. An example of use in autonomous vehicles is the Kalman Filter, which is used to predict movement and improve vehicle position estimates based on IMU data. In robotics, this algorithm supports more precise navigation and control, while in augmented reality applications, Kalman Filter helps integrate IMU sensor data with 3D visualization to make it more realistic. The Gy-91 sensor is a complex module of an Inertial Measurement Unit (IMU) sensor, which contains 9-dofs with 3-dof accelerometers, 3-dof gyroscopes, and 3-dof magnetometers. This sensor measures acceleration, angular speed, and orientation of objects in three dimensions (3D). However, acceleration and speed data from IMU sensors are often subject to interference, noise, and drift, so processing methods such as the Kalman Filter algorithm are needed to predict and correct more accurate sensor values.

Kalman Filter is a recursive statistical algorithm suitable for processing IMU sensor data. With the Kalman Filter, sensor data from the Gy-91 can be analyzed and filtered, resulting in more precise estimates of orientation, acceleration, and angular velocity. The Kalman Filter process consists of two main stages: prediction and update. The Kalman filter algorithm is an algorithm that can be applied to nonlinear systems. This becomes a challenge if the nonlinear system is run on a drone. To predict the tilt value of the drone in a 3D area, KF combines a system state model and noise measurements to obtain optimal data. The research [3] used the extended Kalman filter method to optimize the prediction value. The Extended Kalman filter algorithm develops the basic Kalman filter model. Of course, if you use a development model, the algorithm's performance will increase because the calculation

formula will also increase. Thus, in this research, the Kalman filter is maintained in the basic model with an approach of changing the constant of the Measurement error, Estimate error, and Covarian Process on program code to find the optimal constant. A Kalman filter is applied to emulate the IMU sensor data of the GY-91 module. The GY-91 sensor has a 9-axis output, which then uses a fusion formula with a Madgwick filter to produce data output in the form of a 3-axis on each of the pitch, roll, and yaw axes. The ARM STM32F407VGT6 pin reads sensor data, and the Kalman Filter process is carried out. In this research, applying the Kalman Filter to the IMU sensor produces more accurate estimates, allowing filter technology to predict the movement and orientation of objects better and more precisely.

II. METHOD

A. Sensor Inertial Measurement Unit (IMU)

The GY-91 board, as seen in Fig.1, is a type of IMU sensor module that has 9-axis output at once, each of which consists of a 3-axis Gyroscope sensor, 3-axis Accelerometer sensor, and 3-axis magnetometer sensor - found on the MPU-9250 (also equipped with a temperature sensor). In addition to the MPU-9250 chip, the GY-91 is equipped with a BMP-280 pressure sensor (which also has a temperature sensor) and a digital Compass sensor. This tiny 9-axis GY-91 sensor module is more complex than the GY-80 module. This module combines a single-chip MPU-9250 with a Gyroscope, Accelerometer, Magnetometer, Digital Compass, and BMP280 pressure sensor. The MPU-9250 is a 9-axis motion detection device. The MPU-9250 comes in a 3x3x1mm QFN package, enabling significant chip size and power consumption reductions while improving performance and cost. Improvements include supporting the accelerometer low-power mode at just 6.4 μ A and improving compass data resolution of 16-bit (0.15 μ T per LSB). A full-scale measuring range of $\pm 4800\mu$ T helps alleviate the challenges of compass placement on complex PCBs. This IMU module is a small board that holds all the sensors needed for 9 degrees of freedom. This module communicates with Arduino or microcontroller via the I2C interface.

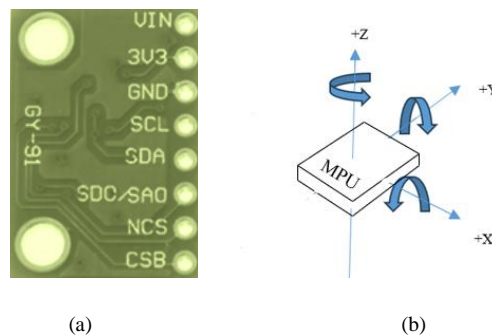


Fig. 1. Sensor IMU (a) Modul GY-91 (b) The direction of the axis on the sensor MPU9250

The MPU9250 module is a sensor that functions as an accelerometer and a gyroscope packaged in 1 module that is compatible with Arduino. The Accelerometer sensor measures the acceleration of a moving object or object that is dynamic or static. Dynamic measurements are measurements of the acceleration of moving objects, while static measurements are measurements of the acceleration of objects relative to Earth's gravity. For this reason, accelerometers are more often used to measure tilt angles, and basically, the working principle of an accelerometer is to use the principle of acceleration. A gyroscope is a sensor that determines movement orientation by relying on a wheel that rotates rapidly on an axis based on angular momentum. To access the MPU9250 program using the Arduino Manager Library, use a library such as Hideakitai MPU9250 in Fig.2.

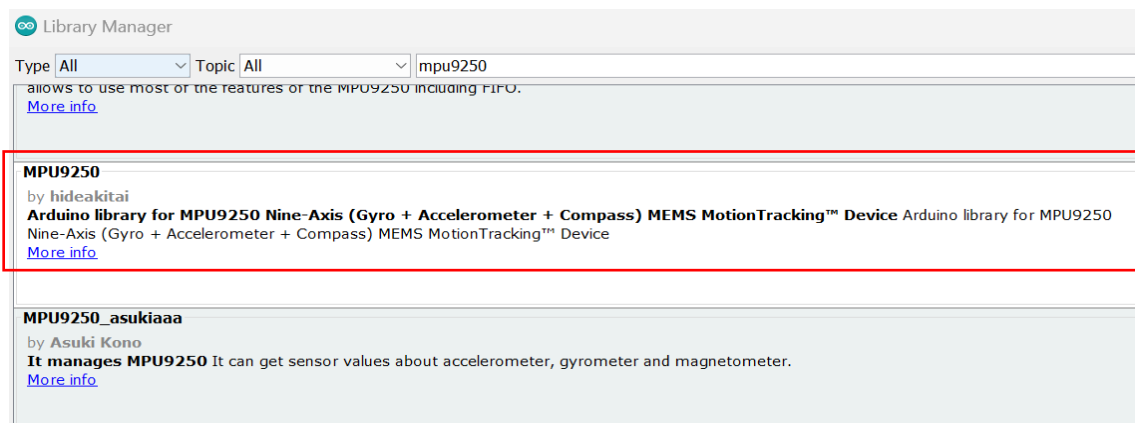


Fig. 2. MPU9250 Arduino Library Manager

B. Kalman Filter Algorithm

Kalman Filter in Fig.3 is a statistical algorithm used to estimate and predict values related to a system. This algorithm has various applications, one in two dimensions, where estimation and prediction are carried out in two parameters or variables. The 2D Kalman Filter is very useful in various contexts, including navigation, object tracking, and image processing. Applying the Kalman Filter in two dimensions involves mathematical modeling of a system involving two interrelated variables. In each iteration, the Kalman Filter algorithm performs two main stages:

- Prediction: The Kalman Filter algorithm estimates sensor values based on the system's mathematical model and previous estimates. The algorithm predicts future IMU sensor values based on expected speed, acceleration, and orientation changes.
- Update: The update stage uses the measurement data provided by GY-91 to update previous predictions. This measurement data is used to improve predictions and produce more accurate estimates.

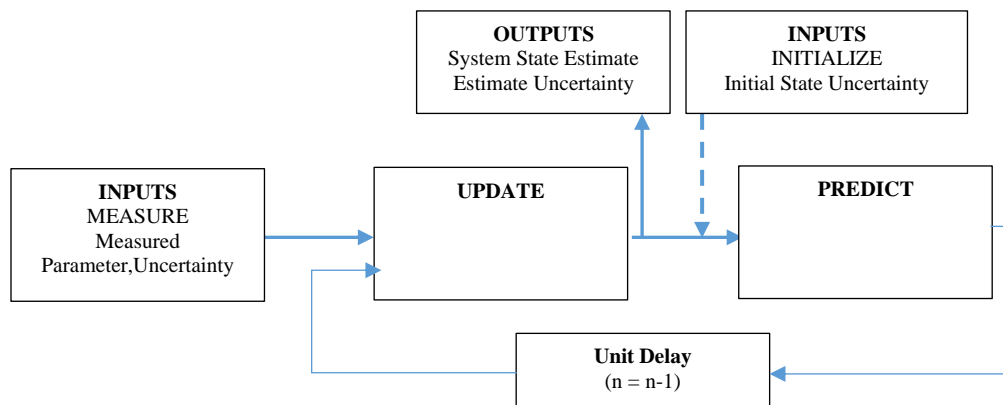


Fig. 3. Kalman Filter Design

Kalman Filter is an algorithm used to combine measurement data with a mathematical model of the system to predict and update estimated values associated with the system as in Fig. 4. In Equation (1) is the Kalman Filter formula along with an explanation of each step. Where the \hat{X}_k variable is a current estimation, the K_k variable is Kalman gain, the Z_k variable is measured value, and the \hat{X}_{k-1} variable is the previous estimation. Step prediction is carried out to calculate the estimated state value at the next time, while step correction is carried out to smooth out and find the optimal value of Kalman Gain using the measured value at that time.

$$\hat{X}_k = K_k \cdot Z_k + (1 - K_k) \cdot \hat{X}_{k-1} \quad (1)$$

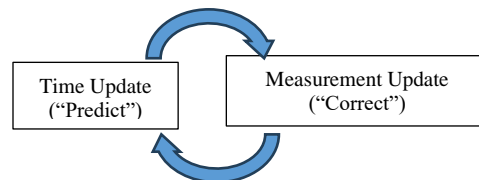


Fig. 4. Kalman Filter Update Process

C. Hardware

As a means of accessing data from the GY-91 sensor module using the ARM STM32F407VGT6 microprocessor and implemented in the EFALCON minimum system as in Fig. 5. EFALCON is a minimum system module consisting of the ARM IC and GY-91 sensor, this simple module is used for drone flight controllers.

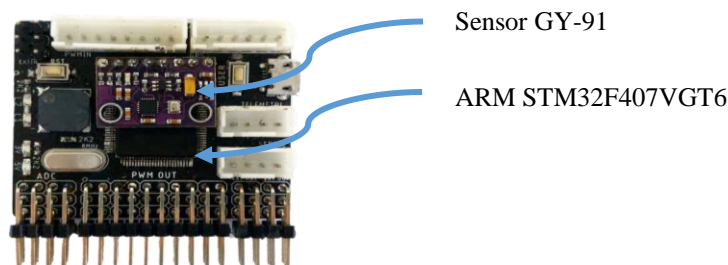


Fig.5. Minimum System with ARM STM32F407VGT6

The hardware circuit on the ARM STM32F407VGT6 with a 168MHz clock is in Fig.6.

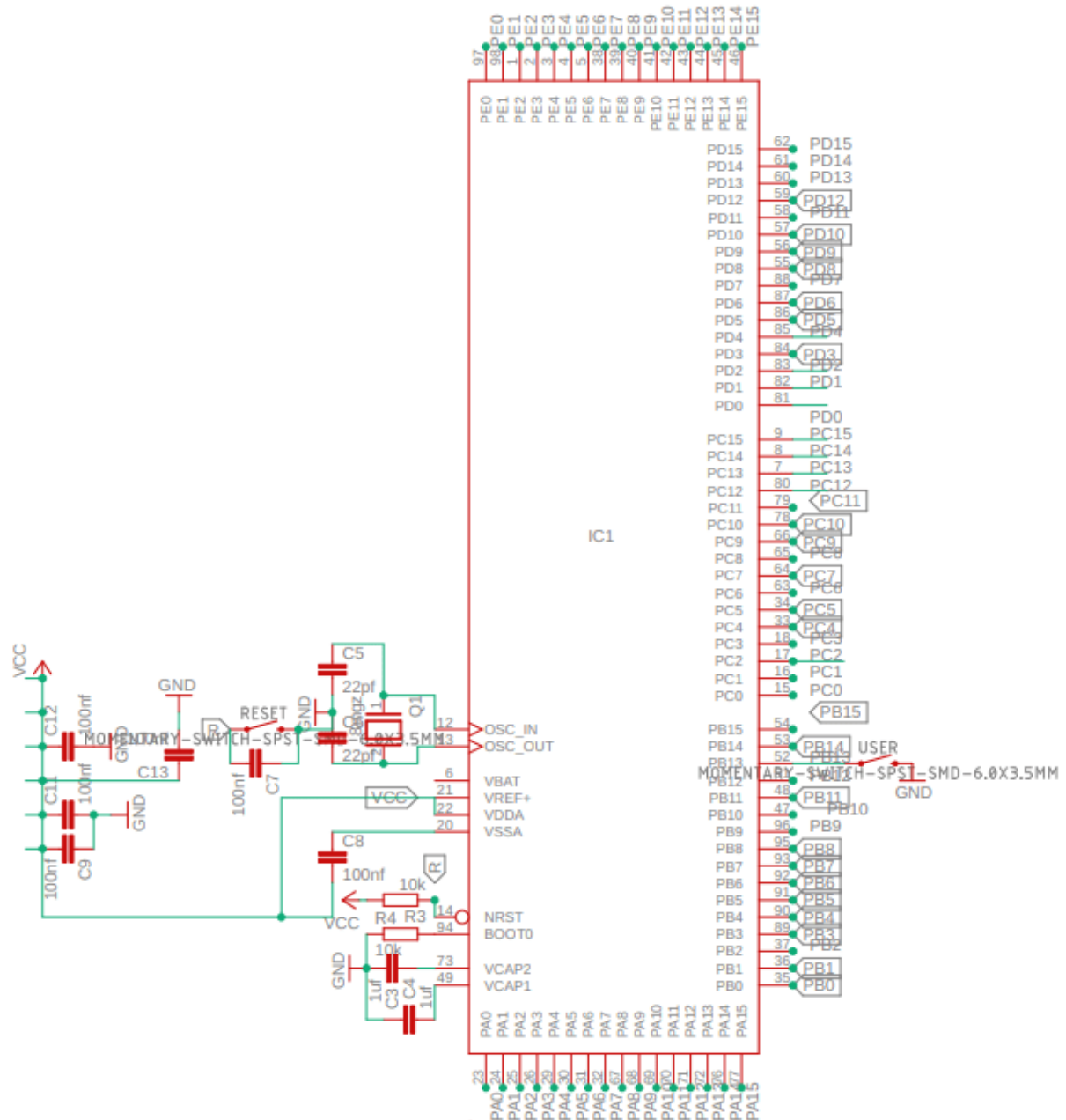


Fig. 6. ARM STM32F407VGT6 Hardware Schematic Connection

In order to gain access to the GY-91 IMU sensor using the ARM STM32F407VGT6 module, the IMU sensor must first be linked to a microcontroller. The pin configuration must be altered to match the one in Table 1. It is necessary to implement algorithms such as sensor fusion (such as the Madgwick filter) to combine data from various IMU sensors originating from the GY-91 sensor. The IMU sensor may require calibration to produce accurate data. This usually involves measuring the offset value and compensating for sensor drift—connection GY-91 pin schematic as in Fig.7.

GY-91	ARM STM32F407V
SCL	SCK
SDA	MOSI
SAO	MISO
GND	GND
3.3V	3.3V
5V	5V

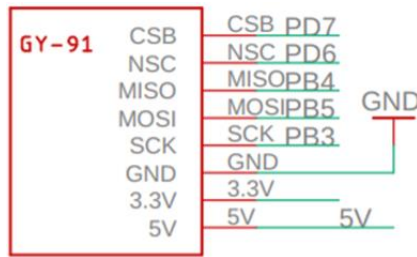


Fig.7. Connection with GY-91 to ARM STM32F407VGT6 Hardware Schematic

D. Design System

The block diagram for estimating the GY-91 sensor value is in Fig.8

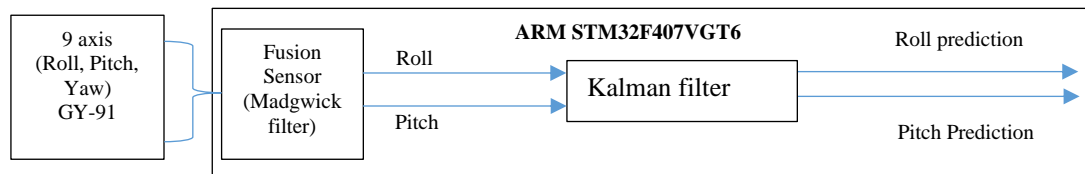


Fig.8. Design System

The explanation of diagram Fig.8 is that the sensor readings used in this research are only roll and pitch data from the 9-axis fusion sensor output GY-91. The Kalman filter block has a process for predicting roll and pitch sensor values. The value of this sensor needs to be predicted because a lot of data is disturbed by noise. Prediction results are used to update sensor reading data. On the ARM STM32F407VGT6, an SPI peripheral can be used to communicate with the MPU9250 sensor. The data is combined using the Madgwick filter [10] to obtain the orientation of the yaw, pitch, and roll angles. Madgwick filter is a sensor fusion method that combines data from accelerometer, gyroscope, and magnetometer into estimated data on the orientation angle of an object. These angle estimates can be quaternions or Euler angles. Fig.9 explains the block diagram of the Madgwick Filter.

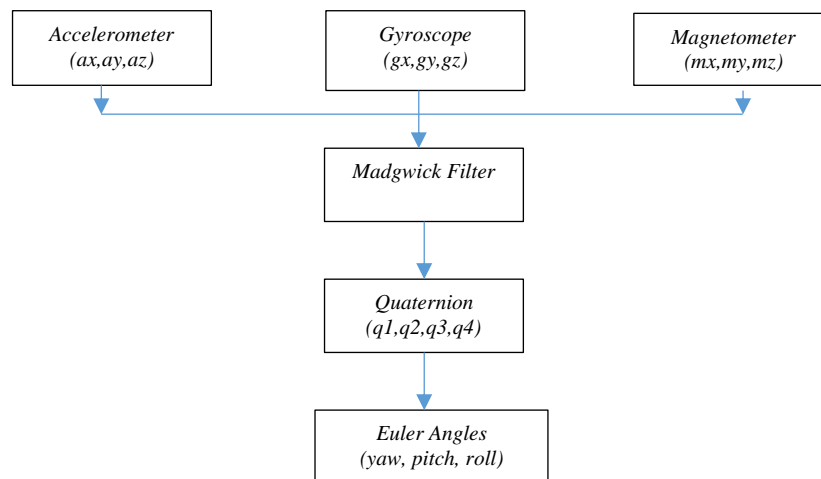


Fig.9. Fusion Sensor

The program code for sensor fusion is as follows:

For Fusion Gyroscope

```
const FusionVector3 halfGravity = {
    .axis.x = Q.x * Q.z - Q.w * Q.y,
    .axis.y = Q.w * Q.x + Q.y * Q.z,
    .axis.z = Q.w * Q.w - 0.5f + Q.z * Q.z,
};
```

For Fusion Magnetometer

```
const FusionVector3 halfWest = {
    .axis.x = Q.x * Q.y + Q.w * Q.z,
```

```
.axis.y = Q.w * Q.w - 0.5f + Q.y * Q.y,  
.axis.z = Q.y * Q.z - Q.w * Q.x  
};
```

For Fusion Accelerometer

```
const FusionVector3 gravity = {  
.axis.x = 2.of * (Q.x * Q.z - Q.w * Q.y),  
.axis.y = 2.of * (Q.w * Q.x + Q.y * Q.z),  
.axis.z = 2.of * (Q.w * Q.w - 0.5f + Q.z * Q.z),  
};
```

The sensor fusion results from 9-axis to 3-axis roll, pitch, and yaw. However, the data used in this research is only roll and pitch.

III. RESULT AND DISCUSSION

Several parameters were carried out; namely, time sampling for each data collection resulting from sensor readings was carried out every 20ms. Then, the variable parameters in the Kalman filter are measurement error, estimation error, and covariance error. The value of each variable ranges between 0-1. The program's variable layout is in Fig.10. These changes can be changed directly via program code because the Kalman filter program embedded here uses a library.

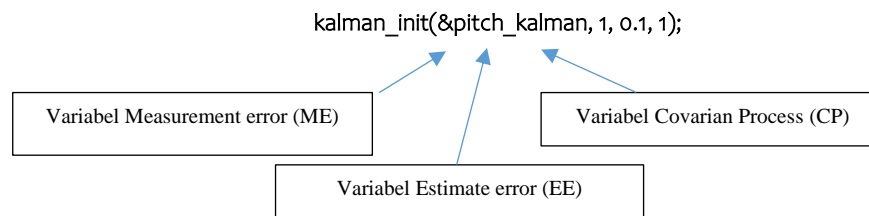


Fig.10. Change Variable Kalman Filter In Program Code

With 300 sampling data readings, the Kalman Filter (KF) results were obtained as follows:

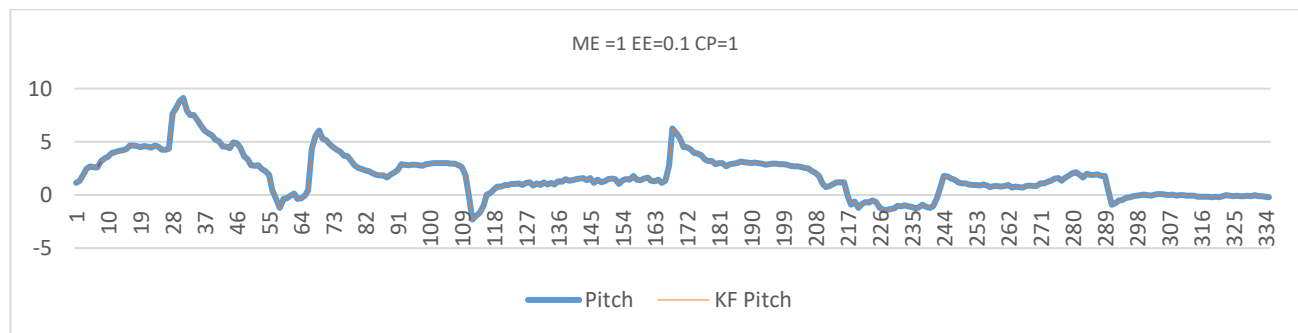


Fig.11. KF-Pitch data with ME =1 EE=0.1 CP=1

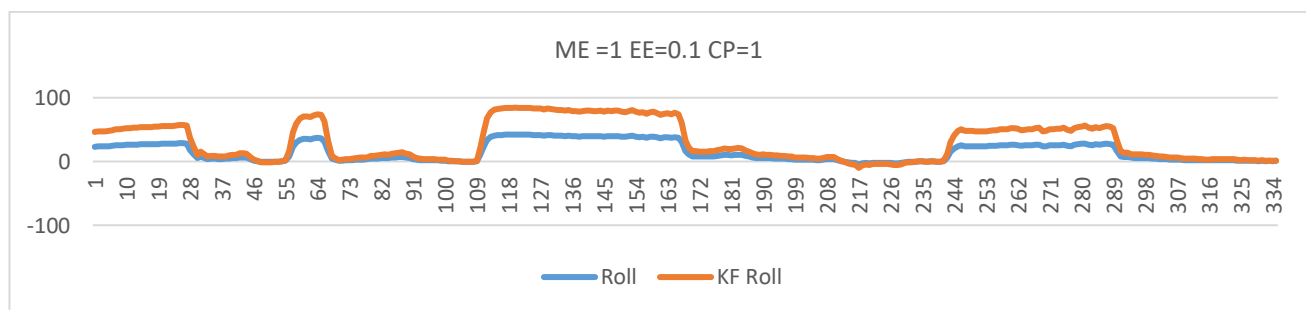


Fig.12. KF-Roll data with ME =1 EE=0.1 CP=1

In the experimental results, according to Fig. 11, the orange graph is the result of the Kalman filter for pitch data with variable parameters with ME = 1 EE = 0.1 CP = 1. The filter results have succeeded in predicting the pitch value. Meanwhile, with the same parameters in Fig.12, the Kalman filter results from the prediction roll data are inappropriate.

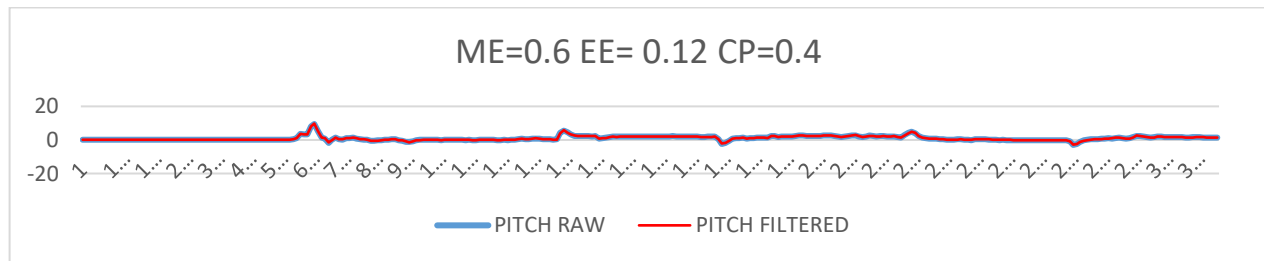


Fig.13. KF-Pitch data with ME =0.6 EE=0.12 CP=0.4

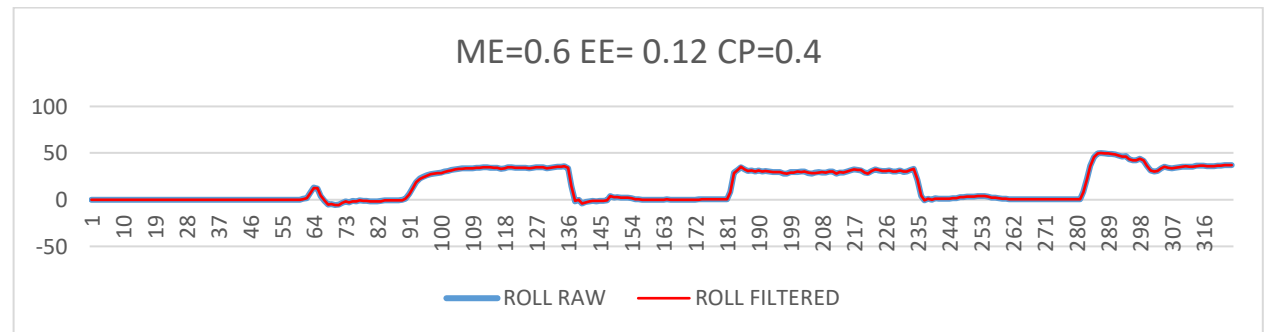


Fig.14. KF-Roll data with ME =0.6 EE=0.12 CP=0.4

In the experimental results, according to Fig.13, the red graph is the result of the Kalman filter for pitch data with variable parameters with ME = 0.6 EE = 0.12 CP = 0.4. The filter results have succeeded in predicting the pitch value. In Fig.14, the results of the Kalman filter from the prediction roll data are also suitable.

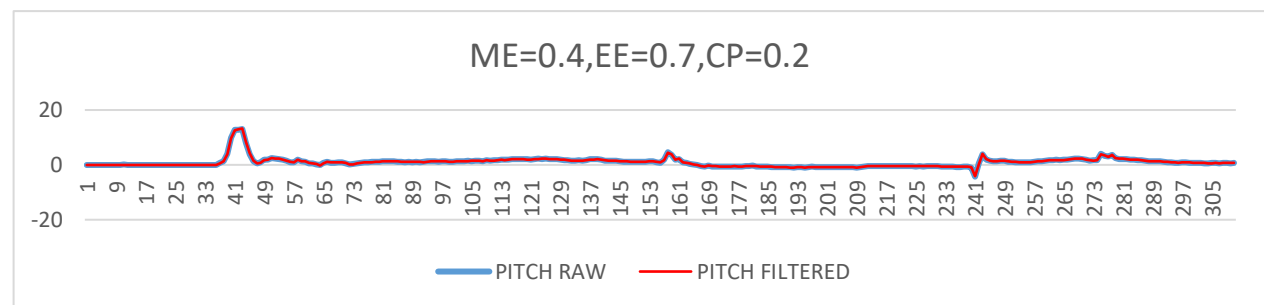


Fig.15. KF-Pitch data with ME=0.4, EE=0.7, CP=0.2

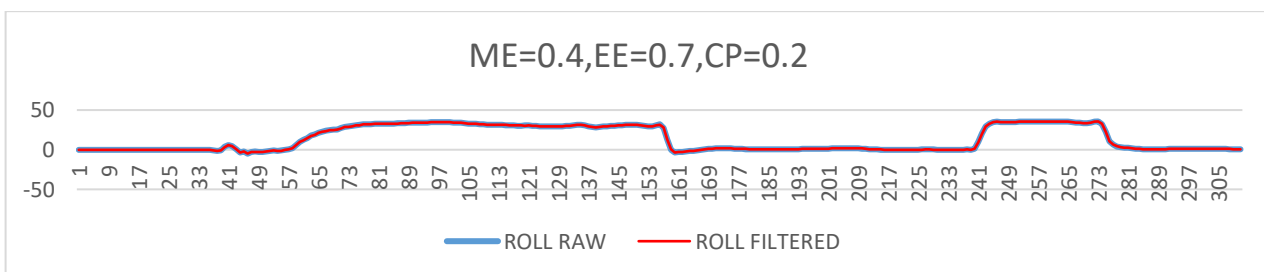


Fig.16. KF-Roll data with ME=0.4, EE=0.7, CP=0.2

In the experimental results according to Fig. 15, the red graph is the result of the Kalman filter for pitch data with variable parameters with ME = 0.6 EE = 0.12 CP = 0.4. The filter results have succeeded in predicting the pitch value. In Fig.16, the results of the Kalman filter from the prediction roll data are also suitable. Kalman filter prediction from pitch data Fig. 15 and Fig. 16 can be estimated correctly. Still, in Fig. 16, with an estimated error of 0.7, some predictions are incorrect, especially if changes in data values change dynamically. The Kalman filter with ME=1, EE=0.12, and CP=0.4 provides the most appropriate values when estimating data from the IMU sensor. This means that the basic algorithm concept of the Kalman filter can be used to estimate and predict IMU data. Apart from that, with the basic algorithm, the calculation process is faster than the modified Kalman filter algorithm, such as the extended Kalman filter.

IV. CONCLUSION

From the observation results, it is known that there are pitch and roll data values in the positive dimension (positive data values) and also in the negative dimension (negative data values). This indicates that it is a rotational movement of the axis. If the data value is positive, the sensor moves to the right. The sensor moves to the left if the data value is negative. The test results prove that the Kalman filter with values $ME=1$, $EE=0.12$, and $CP=0.4$ provides the most appropriate values when estimating the Kalman filter value. With a combination of these constant values, the Kalman filter can predict pitch and roll values optimally so that when applied to a nonlinear system, namely to a drone when flying, for example, this variable constant approach can predict the tilt value of the drone in a 3D area. The filter can still predict significant changes in pitch and roll values.

ACKNOWLEDGMENTS

Thank you to the Politeknik Elektronika Negeri Surabaya for funding this research. The funding scheme used is Local Research. This scheme has a big role in fostering innovations for lecturers in their work.

REFERENCES

- [1] K. Kefferpütz and K. McGuire, "Error-State Unscented Kalman-Filter for UAV Indoor Navigation," 2022 25th International Conference on Information Fusion (FUSION), Linköping, Sweden, 2022, pp. 01-08, doi: 10.23919/FUSION49751.2022.9841385.
- [2] P. Somers, J. Schüle, C. Tarín and O. Sawodny, "2D to 3D Segmentation: Inclusion of Prior Information using Random Walk Kalman Filters," 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Mexico, 2021, pp. 4222-4225, doi: 10.1109/EMBC46164.2021.9629734.
- [3] C. T. Ginalih, A. S. Jatmiko and R. Darmakusuma, "Simple Application of Kalman Filter On a Moving Object in Unity3D," 2020 6th International Conference on Interactive Digital Media (ICIDM), Bandung, Indonesia, 2020, pp. 1-3, doi: 10.1109/ICIDM51048.2020.9339662.
- [4] A. Eman and H. Ramdane, "Mobile Robot Localization Using Extended Kalman Filter," 2020 3rd International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia, 2020, pp. 1-5, doi: 10.1109/ICCAIS48893.2020.9096805.
- [5] Y. Luo, G. Ye, Y. Wu, J. Guo, J. Liang and Y. Yang, "An Adaptive Kalman Filter for UAV Attitude Estimation," 2019 IEEE 2nd International Conference on Electronics Technology (ICET), Chengdu, China, 2019, pp. 258-262, doi: 10.1109/ELTECH.2019.8839496.
- [6] A. P. Dhongade and M. A. Khandekar, "GPS and IMU Integration on an autonomous vehicle using Kalman filter (LabView Tool)," 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 2019, pp. 1122-1125, doi: 10.1109/ICCS45141.2019.9065851.
- [7] Z. Li, Z. Su and T. Yang, "Design of Intelligent Mobile Robot Positioning Algorithm Based on IMU/Odometer/Lidar," 2019 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC), Beijing, China, 2019, pp. 627-631, doi: 10.1109/SDPC.2019.00118.
- [8] L. Lasmadi, F. Kurniawan, D. Dermawan and G. N. P. Pratama, "Mobile Robot Localization via Unscented Kalman Filter," 2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, 2019, pp. 129-132, doi: 10.1109/ISRITI48646.2019.9034570.
- [9] J. Onshaunjit and J. Srinonchat, "Nonlinear Kalman Filter Model for Balancing Optimization Gyro Robot," 2018 International Electrical Engineering Congress (iEECON), Krabi, Thailand, 2018, pp. 1-4, doi: 10.1109/IEECON.2018.8712267.
- [10] S. Habbachi, M. Sayadi, F. Fnaiech, N. Rezzoug, P. Gorce and M. Benbouzid, "Estimation of IMU orientation using linear Kalman Filter based on correntropy criterion," 2018 IEEE International Conference on Industrial Technology (ICIT), Lyon, France, 2018, pp. 1340-1344, doi: 10.1109/ICIT.2018.8352373.
- [11] H. C. Kam, Y. K. Yu and K. H. Wong, "An Improvement on ArUco Marker for Pose Tracking Using Kalman Filter," 2018 19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Busan, Korea (South), 2018, pp. 65-69, doi: 10.1109/SNPD.2018.8441049.
- [12] H. . -Q. . -T. Ngo, T. . -P. Nguyen, V. . -N. . -S. Huynh, T. . -S. Le and C. . -T. Nguyen, "Experimental comparison of Complementary filter and Kalman filter design for low-cost sensor in quadcopter," 2017 International Conference on System Science and Engineering (ICSSE), Ho Chi Minh City, Vietnam, 2017, pp. 488-493, doi: 10.1109/ICSSE.2017.8030922.
- [13] A. Kaba, A. Ermeydan and E. Kiyak, "Model derivation, attitude control and Kalman filter estimation of a quadcopter," 2017 4th International Conference on Electrical and Electronic Engineering (ICEEE), Ankara, Turkey, 2017, pp. 210-214, doi: 10.1109/ICEEE2.2017.7935821.
- [14] M. Badeche and M. Benmohamed, "Real-Time Tracking for Augmented reality," 2006 2nd International Conference on Information & Communication Technologies, Damascus, Syria, 2006, pp. 1773-1778, doi: 10.1109/ICTTA.2006.1684654.