

Implementasi Algoritma 2 Step Kalman Filter Untuk Mengurangi Noise Pada Estimasi Data Accelerometer

Wahyu Sukestyastama Putra
Universitas AMIKOM Yogyakarta
Jl. Ring Road Utara, Condong Catur, Depok, Sleman, Yogyakarta
wahyu@amikom.ac.id

Abstract

An accelerometer is a useful sensor in technological development. Currently, the accelerometer is found on smartphone devices, navigation devices, and wearable devices. However, processing the sensor output signal into data that can be interpreted is not easy. This is because the output of an accelerometer sensor has significant noise. In this study, the authors are interested in developing an estimation method using a Kalman Filter. Kalman filter is an estimator so it is expected that the sensor data are more resistant to noise interference. In this study, the author innovated the 2 step Kalman filter. The study was conducted because the use of 1 step still has noise on the estimation results. Based on the analysis of the algorithm simulation results, it can be concluded that the Kalman filter 2-step algorithm has good performance in estimating the accelerometer sensor output. When compared with the Kalman filter 1 step algorithm, the Kalman filter 2 step algorithm has a smaller average error estimation and is able to achieve a constant/stable condition faster than the Kalman filter 1 step method

Keywords: Accelerometer, Kalman Filter, Noise, State

Abstrak

Accelerometer adalah sensor yang memiliki banyak manfaat dalam perkembangan teknologi. Saat ini accelerometer banyak di jumpai pada perangkat smartphone, alat navigasi maupun wearable device. Akan tetapi pengolahan sinyal output sensor menjadi data yang dapat diinterpretasikan tidak mudah. Ini dikarenakan output sensor accelerometer memiliki noise yang cukup signifikan. Dalam penelitian ini penulis tertarik untuk mengembangkan metode estimasi menggunakan Kalman Filter. Kalman filter merupakan sebuah estimator sehingga diharapkan hasil prediksi output sensor lebih tahan terhadap gangguan noise. Pada penelitian ini penulis melakukan inovasi 2 step Kalman filter. Penelitian dilakukan karena penggunaan 1 step masih memiliki noise pada hasil estimasi. Berdasarkan analisis hasil simulasi algoritma yang dilakukan, dapat disimpulkan bahwa algoritma 2 step Kalman filter memiliki performa yang baik dalam mengestimasi output sensor accelerometer. Jika dibandingkan dengan algoritma 1 step Kalman filter, algoritma 2 step Kalman filter memiliki error rata rata estimasi yang lebih kecil dan mampu mencapai kondisi konstan/stabil lebih cepat daripada metode 1 step Kalman filter.

Kata kunci: Accelerometer, Kalman Filter, Noise, State

1. PENDAHULUAN

Accelerometer adalah sensor yang memiliki banyak manfaat dalam perkembangan teknologi. Saat ini accelerometer banyak di jumpai pada perangkat smartphone, alat navigasi maupun *wearable device*. Inovasi berbasis accelerometer yang saat ini dikembangkan diantaranya aplikasi

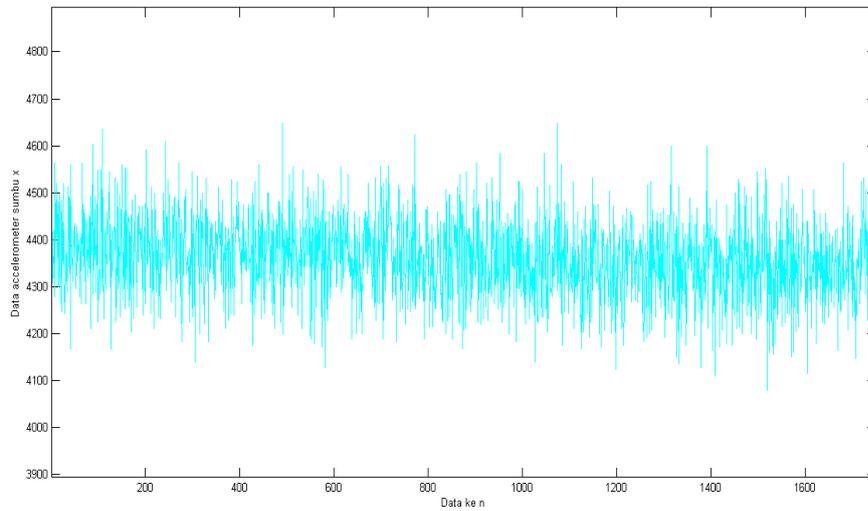
perangkat mobile [1], perangkat kesehatan [2-3], navigasi maupun autopilot [4-6]. Akan tetapi pengolahan sinyal output sensor menjadi data yang dapat diinterpretasikan tidak mudah. Ini dikarenakan output sensor accelerometer memiliki noise yang cukup signifikan. Dengan demikian diperlukan suatu cara untuk mengekstrak data dari sinyal yang memiliki noise. Banyak metode yang sebelumnya telah dilakukan untuk mengolah data accelerometer. Metode tersebut diantaranya adalah metode *complementary filter* [7], Kalman Filter [8], Neural Network dan Algoritma Genetika [9]. Dalam penelitian ini penulis tertarik untuk mengembangkan metode estimasi menggunakan Kalman Filter. Kalman filter merupakan sebuah estimator sehingga diharapkan hasil prediksi output sensor lebih tahan terhadap gangguan noise.

Perbedaan penelitian ini dengan penelitian sebelumnya terletak pada step filtering yang digunakan. Pada penelitian sebelumnya [10-11], step filtering yang digunakan adalah 1 step Kalman Filter. Pada penelitian ini penulis tertarik melakukan penelitian 2 step Kalman filter untuk mengekstrak data dari accelerometer. Penelitian ini dilakukan karena penggunaan 1 step Kalman Filter masih memiliki noise pada hasil estimasi. Dengan adanya inovasi tersebut diharapkan hasil ekstrak data dapat semakin baik sehingga informasi yang diberikan lebih akurat.

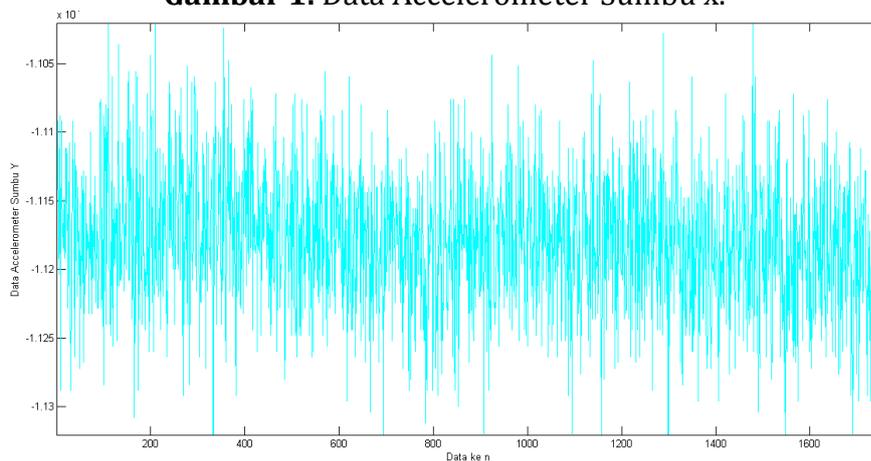
2. METODOLOGI PENELITIAN

Metode penelitian yang dilakukan adalah metode eksperimen. Eksperimen dilakukan dengan mengimplementasikan algoritma yang dirancang pada data output accelerometer. Data hasil estimasi algoritma 2 step Kalman filter kemudian dibandingkan dengan data output estimasi algoritma 1 step Kalman filter sebagai pembanding. Hasil yang dianalisis adalah error estimasi masing masing algoritma dan kecepatan masing masing algoritma untuk mencapai kondisi stabil/konstan pada masing masing sumbu accelerometer.

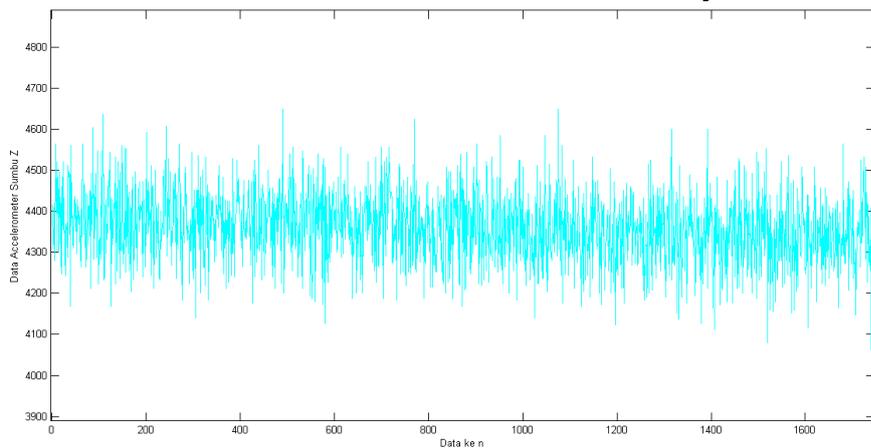
Data dalam penelitian diambil menggunakan sensor accelerometer MPU6050 dalam kondisi diam. Data sensor kemudian dibaca oleh mikrokontroler Arduino uno menggunakan komunikasi data I2C. Data hasil pembacaan accelerometer kemudian dikirim ke komputer menggunakan komunikasi serial 115200 Bps. Data yang direkam adalah data accelerometer sumbu x, sumbu y dan sumbu x. Data tersebut kemudian di log dengan interval 0.25 detik. Data accelerometer tersebut berupa bilangan integer yang memiliki banyak noise sehingga perlu difilter terlebih dahulu. Data output accelerometer yang digunakan digambarkan pada Gambar 1, Gambar 2 dan Gambar 3.



Gambar 1. Data Accelerometer Sumbu x.



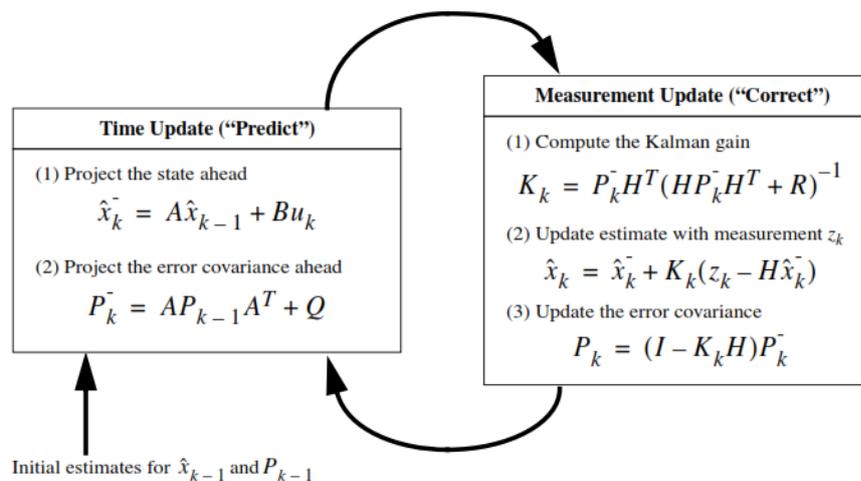
Gambar 2. Data Accelerometer Sumbu y.



Gambar 3. Data Accelerometer Sumbu z.

Kalman filter adalah sebuah rangkaian persamaan matematis yang memiliki kemampuan untuk mengestimasi state suatu proses secara recursif dan

efisien dengan cara meminimalkan rata rata error kuadrat [12]. Kalman filter memiliki keunggulan dalam melakukan estimasi state meskipun karakteristik sistem tidak diketahui secara pasti (*unknown*). Kalman filter memiliki 2 tahap proses yaitu proses prediksi dan proses koreksi. Proses prediksi adalah proses memperkirakan kondisi state dan error model berdasarkan kondisi state dan error sebelumnya. Sedangkan proses koreksi adalah proses penyesuaian state dan error prediksi berdasarkan kondisi terkini. Diagram proses algoritma Kalman filter digambarkan pada Gambar 4.



Gambar 4. Diagram Proses Algoritma Kalman Filter [12]

Untuk mengimplementasikan algoritma Kalman Filter pada accelerometer memerlukan penyesuaian terhadap kondisi yang ada. Penyesuaian meliputi:

1. Memprediksi State.

Pada tahap ini penyesuaian dilakukan dengan memberikan nilai $A=1$ dan menghilangkan komponen input sistem (Bu_k) karena sistem yang kita gunakan tidak memiliki input. Pada sistem ini kita hanya melakukan pengamatan kondisi state. Dengan demikian persamaan yang digunakan untuk memprediksi state adalah,

$$\hat{x}_k^- = \hat{x}_k. \quad (1)$$

Persamaan (1) kemudian diterjemahkan ke bahasa pemrograman menjadi:

$$\text{ESTprev} = \text{ESTcurr}$$

2. Memprediksi Error.

Pada tahap ini penyesuaian dilakukan dengan memberikan nilai $A=1$ dan menghilangkan komponen *Noise Covarian* (Q) karena sistem yang kita gunakan hanya sistem 1 variabel. Dengan demikian persamaan yang digunakan untuk memprediksi Error adalah,

$$P_k^- = P_k. \quad (2)$$

Persamaan (2) kemudian diterjemahkan ke bahasa pemrograman menjadi:

$$\text{ERRprev} = \text{ERRcurr}$$

3. Menghitung Kalman Gain

Pada tahap ini penyesuaian dilakukan dengan memberikan nilai $H=1$ dan menambahkan komponen *Noise* pengukuran (R). H adalah matriks pengukuran, sedangkan R adalah error pengukuran. Dengan demikian persamaan yang digunakan untuk menghitung Kalman Gain adalah,

$$K_k = \frac{P_k^-}{P_k^- + R}. \quad (3)$$

Persamaan (3) kemudian diterjemahkan ke bahasa pemrograman menjadi:

$$\text{KG} = \text{ERRprev} / (\text{ERRprev} + \text{ERRmea})$$

4. Memperbarui nilai State berdasarkan hasil pengukuran

Pada tahap ini penyesuaian dilakukan dengan memberikan nilai $H=1$. Nilai prediksi State pada persamaan (1) dan nilai Kalman Gain pada persamaan (3) digunakan untuk memperbarui nilai State berdasarkan hasil pengukuran menggunakan persamaan,

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - \hat{x}_k^-). \quad (4)$$

Dengan z adalah nilai output sensor accelerometer. Persamaan (4) kemudian diterjemahkan ke bahasa pemrograman menjadi:

$$\text{ESTcurr} = \text{ESTprev} + \text{KG} * (\text{MEA} - \text{ESTprev})$$

5. Memperbarui nilai Error

Pada tahap ini penyesuaian dilakukan dengan memberikan nilai $H=1$. Nilai prediksi Error pada persamaan (2) dan nilai Kalman Gain pada persamaan (3) digunakan untuk memperbarui nilai Error menggunakan persamaan,

$$P_k = (1 - K_k)P_k^-. \quad (5)$$

Persamaan (5) kemudian diterjemahkan ke bahasa pemrograman menjadi:

$$\text{ERRcurr} = (1 - \text{KG}) * \text{ERRprev}$$

Untuk mengimplementasikan 2 Step Kalman Filter terlebih dahulu harus ditentukan nilai inisialisasi. Inisialisasi bertujuan untuk memberikan nilai awal karena nilai sesungguhnya belum didapatkan. Variable yang memiliki nilai inisialisasi adalah:

1. Nilai State sebelum (ESTprev)
2. Nilai Kalman Gain (KG)
3. Nilai State Sekarang (ESTcurr)
4. Nilai Error sekarang (ERRcurr)
5. Nilai Error sebelum (ERRprev)
6. Nilai error pengukuran (ERRmea).

Karena Kalman filter bersifat rekursif, nilai inialisasi tersebut akan berubah mengikuti nilai koreksi yang dilakukan oleh algoritma. Implementasi algoritma 2 step Kalman Filter ke dalam bentuk Pseudocode dijelaskan sebagai berikut.

```
Inialisasi ESTprev
Inialisasi Kalman Gain
Inialisasi ESTcurr
Inialisasi ERRcurr
Inialisasi ERRprev
Inialisasi ERRmea

Mulai iterasi
    Membaca input sensor
    Menghitung KG step 1
    Memprediksi ESTcurr step 1
    Memperbarui nilai ERRcurr step 1
    Update variable ESTprv dan ERRprev step 1

    Jika telah memasuki iterasi ke 3
        Inialisasi ESTprv step 2
        Inialisasi ERRprv step 2
        Membaca ESTcurr Step 1
        Menghitung KG Step 2
        Memprediksi ESTcurr step 2
        Memperbarui nilai ERRcurr step 2
        Update variable ESTprv dan ERRprev step 2

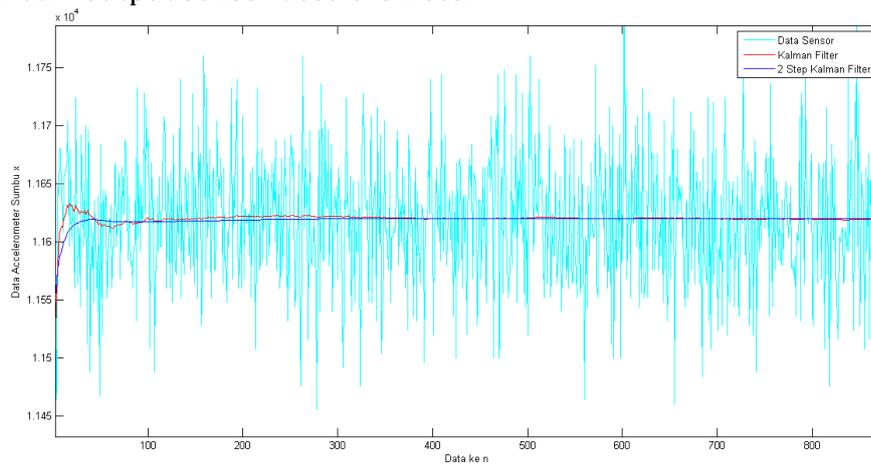
    Selesai
Selesai
```

Pada algoritma yang dirancang, Kalman filter step kedua dilakukan setelah mencapai iterasi data ke tiga. Kondisi tersebut dilakukan untuk melakukan inialisasi nilai ESTprv2 dan ERRprev2. Hal tersebut dilakukan untuk meminimalisasi kesalahan inialisasi nilai pada proses Kalman filter step kedua. Hasil estimasi pada algoritma yang dirancang adalah nilai dari ESTcurr step 2. Pseudocode tersebut kemudian diterjemahkan pada bahasa pemrograman untuk selanjutnya diimplementasikan pada software Matlab. Software Matlab digunakan untuk melakukan simulasi algoritma yang dirancang. Dari hasil simulasi algoritma tersebut kemudian dilakukan analisis karakteristik algoritma yang dirancang dengan algoritma Kalman filter.

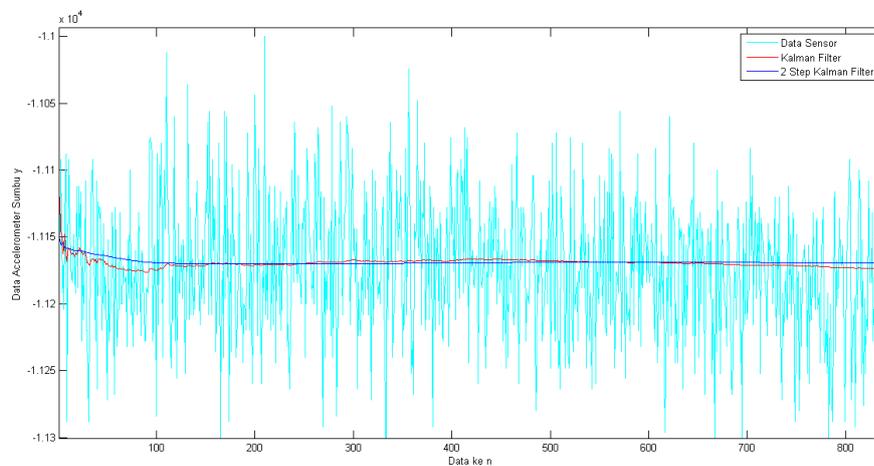
3. HASIL DAN PEMBAHASAN

Hasil simulasi Algoritma menggunakan data accelerometer ditunjukkan pada Gambar 5, 6 dan 7. Hasil simulasi tersebut menunjukkan perbandingan

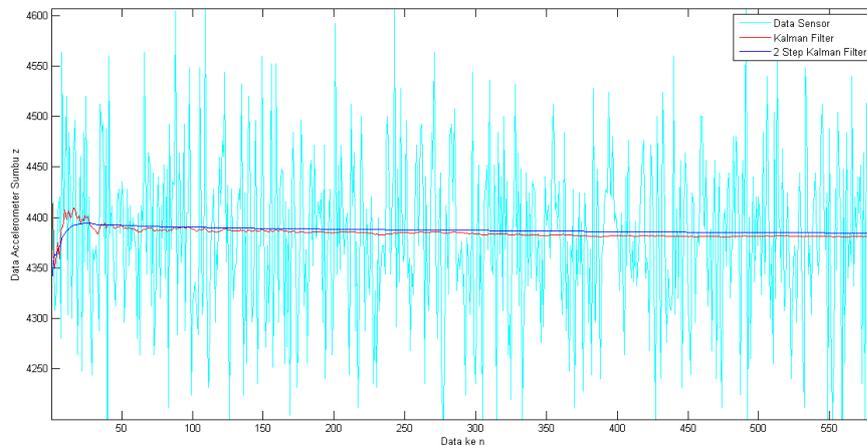
antara data sensor accelerometer yang tidak di filter (warna cyan), hasil estimasi state menggunakan Kalman Filter (warna merah) dan hasil estimasi menggunakan 2 step Kalman Filter (warna biru). Absis pada grafik menunjukkan update data ke n dengan interval pengambilan data 0.25 detik. Dengan informasi tersebut diharapkan dapat mengetahui kestabilan algoritma dalam melakukan estimasi. Ordinat pada grafik menunjukkan nilai integer dari output sensor accelerometer.



Gambar 5. Hasil Fitering Data Accelerometer Sumbu x



Gambar 6. Hasil Fitering Data Accelerometer Sumbu y



Gambar 7. Hasil Fitering Data Accelerometer Sumbu z

Gambar 5 menunjukkan hasil simulasi pada data accelerometer sumbu x. Gambar tersebut menunjukkan bahwa algoritma Kalman filter dapat mengestimasi kondisi accelerometer. Algoritma Kalman filter dapat mencapai kondisi konstan/stabil pada iterasi ke 100. Permasalahan yang ada adalah algoritma Kalman filter mengalami osilasi error sebelum mencapai kondisi stabil tersebut. Sedangkan algoritma 2 step Kalman Filter mencapai kondisi konstan/stabil pada iterasi ke 40. Pada algoritma 2 step Kalman Filter tidak dijumpai osilasi error sebelum mencapai kondisi stabil/konstan. Sedangkan untuk iterasi lebih dari 100, kedua algoritma menunjukkan kondisi yang relatif sama dan stabil

Gambar 6 menunjukkan hasil simulasi pada data accelerometer sumbu y. Gambar tersebut menunjukkan bahwa algoritma Kalman filter dapat mengestimasi kondisi accelerometer dengan baik. Algoritma Kalman filter dapat mencapai kondisi konstan/stabil pada iterasi ke 120. Pada simulasi accelerometer sumbu y yang dilakukan, algoritma Kalman filter mengalami osilasi error sebelum mencapai kondisi stabil tersebut meskipun osilasi tidak setampak pada simulasi accelerometer sumbu x. Sedangkan algoritma 2 step Kalman Filter mencapai kondisi konstan/stabil dengan landai pada iterasi ke 100. Pada algoritma 2 step Kalman Filter tidak dijumpai osilasi error sebelum mencapai kondisi stabil/konstan.

Gambar 7 menunjukkan bahwa hasil simulasi pada data accelerometer sumbu z. Gambar tersebut menunjukkan bahwa algoritma Kalman filter dapat mengestimasi kondisi accelerometer dengan baik. Algoritma Kalman filter dapat mencapai kondisi konstan/stabil pada iterasi ke 400. Permasalahan yang ada adalah algoritma Kalman filter mengalami osilasi error relatif besar pada iterasi kurang dari 50 kemudian mengecil sejak iterasi ke 50 sebelum mencapai kondisi stabil tersebut. Sedangkan algoritma 2 step Kalman Filter mencapai kondisi konstan/stabil pada iterasi ke 20. Pada algoritma 2 step Kalman Filter tidak dijumpai osilasi error sebelum mencapai kondisi stabil/konstan.

Perbandingan hasil simulasi algoritma pada data accelerometer sumbu x, sumbu y dan sumbu z dapat disimpulkan bahwa algoritma 1 step Kalman filter dan 2 step Kalman filter dapat melakukan estimasi state dengan baik. Namun apabila dibandingkan secara karakteristik antara algoritma 2 step Kalman filter dengan 1 step Kalman filter, Algoritma 2 step Kalman filter memiliki performa yang baik jika dibandingkan dengan algoritma 1 step Kalman filter. Hal itu dibuktikan dengan kemampuan algoritma 2 step Kalman filter dalam melakukan estimasi state selalu lebih cepat mencapai kondisi konstan/stabil pada hasil simulasi data accelerometer sumbu x, sumbu y dan sumbu z.

Analisis aspek ketahanan terhadap noise menunjukkan bahwa Algoritma 2 step Kalman filter memiliki performa yang baik jika dibandingkan dengan algoritma 1 step Kalman filter. Hal itu dibuktikan dengan kemampuan algoritma 2 step Kalman filter dalam melakukan estimasi tidak ditemukannya ripple / osilasi error pada hasil simulasi data accelerometer sumbu x, y dan z. Perbandingan error estimasi masing masing metode pada 1000 data awal Accelerometer dirangkum pada Tabel 1.

Tabel 1. Rangkuman Error Estimasi Accelerometer

Metode Estimasi	Sumbu x	Sumbu y	Sumbu z
1 Step Kalman Filter	0.279464148	0.23816101	0.405959849
2 Step Kalman Filter	0.096162674	0.025556872	0.067691207

Tabel 1 menunjukkan perbandingan antara metode 1 step Kalman filter dengan 2 step Kalman filter dalam mengestimasi data accelerometer. Pada sumbu x, algoritma 2 step Kalman filter memiliki rata rata error 1/3 dari metode 1 step Kalman filter. Pada sumbu y, algoritma 2 step Kalman filter memiliki rata rata error sekitar 1/10 dari metode 1 step Kalman filter. Sedangkan pada sumbu z, algoritma estimasi 2 step Kalman filter memiliki rata rata error sekitar 1/6 dari metode 1 step Kalman filter. Dengan melihat performa algoritma estimasi pada masing masing sumbu, dapat disimpulkan bahwa algoritma 2 step Kalman filter lebih baik daripada metode 1 step Kalman filter.

4. SIMPULAN

Berdasarkan analisis yang dilakukan, dapat disimpulkan bahwa algoritma 2 step Kalman filter memiliki performa yang baik dalam mengestimasi output sensor accelerometer. Jika dibandingkan dengan algoritma 1 step Kalman filter, algoritma 2 step Kalman filter memiliki error rata rata estimasi yang lebih kecil dan mampu mencapai kondisi konstan/stabil lebih cepat daripada metode 1 step Kalman filter.

DAFTAR PUSTAKA

- [1] N. M. Rizki and S. Handoko, "Aplikasi Musik Orkestra Angklung Multi Oktaf Berbasis Android Dengan Sensor Accelerometer," *Transformatika*, vol. 15, no. 1, pp. 17–25, 2017.
- [2] T. Bombardini, R. Sicari, E. Bianchini, and E. Picano, "Abnormal shortened diastolic time length at increasing heart rates in patients with abnormal exercise-induced increase in pulmonary artery pressure," *Cardiovasc. Ultrasound*, vol. 9, no. 1, p. 36, 2011.
- [3] W.-Y. Lin, K. V. Verma, M.-Y. Lee, and C.-S. Lai, "Activity Monitoring with a Wrist-Worn, Accelerometer-Based Device," *Micromachines*, vol. 9, no. 9. 2018.
- [4] F. Bonin-Font, M. Massot-Campos, L. P. Negre-Carrasco, G. Oliver-Codina, and P. J. Beltran, "Inertial Sensor Self-Calibration in a Visually-Aided Navigation Approach for a Micro-AUV," *Sensors*, vol. 15, no. 1. 2015.
- [5] C. S. Lee and K. S. Hong, "Velocity-Aided Attitude Estimation for Helicopter Aircraft Using Microelectromechanical System Inertial-Measurement Units," *Sensors*, vol. 16, no. 12. 2016.
- [6] N. Hanafi, "Jurnal Rekayasa ElektriKa," *J. Rekayasa Elektr.*, vol. 14, no. 36, pp. 205–213, 2018.
- [7] R. Maulana, W. Kurniawan, and H. Z. Fahmi, "Noise Reduction on the Tilt Sensor for the Humanoid Robot Balancing System Using Complementary Filter," *MATEC Web Conf.*, vol. 220, 2018.
- [8] Q. Cai, G. Yang, N. Song, and Y. Liu, "Systematic Calibration for Ultra-High Accuracy Inertial Measurement Units," *Sensors*, vol. 16, no. 6. 2016.
- [9] M. Zhu, L. Pang, Z. Xiao, C. Shen, H. Cao, Y. Shi, and J. Liu, "Temperature Drift Compensation for High-G MEMS Accelerometer Based on RBF NN Improved Method," *Applied Sciences*, vol. 9, no. 4. 2019.
- [10] P. Patonis, P. Patias, N. I. Tziavos, D. Rossikopoulos, and G. K. Margaritis, "A Fusion Method for Combining Low-Cost IMU/Magnetometer Outputs for Use in Applications on Mobile Devices," *Sensors*, vol. 18, no. 8. 2018.
- [11] E. Edwan, S. Knedlik, and O. Loffeld, "Angular Motion Estimation Using Dynamic Models in a Gyro-Free Inertial Measurement Unit," *Sensors*, vol. 12, no. 5. 2012.
- [12] G. Welch and G. Bishop, "An Introduction to the Kalman Filter." Chapel Hill:University of North Carolina, pp. 1–16, 2006.