

Jurnal *Rekayasa Elektrika*

VOLUME 13 NOMOR 3

DESEMBER 2017

Pengenal Gerakan dengan Joystick Akselerometer Menggunakan Filter Kalman 172-178

Khoirudin Fathoni dan Dhidik Prastiyanto

JRE	Vol. 13	No. 3	Hal 119–190	Banda Aceh, Desember 2017	ISSN. 1412-4785 e-ISSN. 2252-620X
-----	---------	-------	-------------	------------------------------	--------------------------------------

Pengenalan Gerakan dengan Joystick Akselerometer Menggunakan Filter Kalman

Khoirudin Fathoni dan Dhidik Prastiyanto
Universitas Negeri Semarang
Jl. Sekaran, Sekaran, Gunung Pati, Kota Semarang 50229
e-mail: khoirudinfathoni@mail.unnes.ac.id

Abstrak—Interaksi antara manusia dan mesin semakin berkembang di antaranya melalui pengenalan gerak dengan sensor akselerometer, yang mendeteksi percepatan dari suatu pergerakan. Teknologi ini telah diterapkan oleh perusahaan video game Nintendo melalui *joystick Wiimote* dan *Wiinunchuk* yang saat ini penggunaannya makin luas. Dua permasalahan utama yang muncul pada penggunaan akselerometer adalah bagaimana menghilangkan derau sensor yang dapat mempengaruhi pembacaan data, dan menghilangkan percepatan gravitasi yang terbaca ketika *joystick* pada posisi miring. Derau dan percepatan gravitasi ini dapat mempengaruhi pembacaan data dan akumulasi galat. Artikel ini menjelaskan implementasi Filter Kalman dan teknik sederhana untuk menghilangkan pengaruh percepatan gravitasi untuk mengatasi masalah di atas menggunakan akselerometer dari *joystick WiiNunchuk* pada Arduino Mega 2560. Dari pengujian telah berhasil mereduksi percepatan gravitasi yang muncul saat posisi diam tidak ada gerakan. Pada implementasi Filter Kalman untuk estimasi posisi, kecepatan dan percepatan perlu diperhatikan penentuan parameter q dan R . Saat dibuat R tetap, perubahan nilai q akan menentukan nilai gain K_k yang menentukan lokasi kutub-kutub dari *observer* yang berpengaruh pada kestabilan dan hasil estimasi. Dengan nilai $R = 0,00005$ dan $q=1$, kutub-kutub Filter Kalman terletak di dalam lingkaran satuan sehingga estimasi sistem stabil dan hasilnya dapat mengikuti nilai sensor namun bersih dari derau.

Kata kunci: *Wii-Nunchuk, akselerometer, filter Kalman*

Abstract—Human Machine Interaction keeps growing and developing, one of development is through gesture recognition that detects acceleration in a movement. This technology has been applied in joystick Wiimote and Wii-nunchuk by Nintendo that is widely used all over the world. Two main challenges in using accelerometer are to eliminate the noise of the sensor and to cancel the detected gravity acceleration when the joystick is tilted. The noise and gravity acceleration may influence the data reading and create error accumulation, respectively. This work proposes an implementation of Kalman Filter and also a simple technique to eliminate the influence of the gravity acceleration as a solution to solve above problems in using accelerometer of Wii-Nunchuk joystick in Board Arduino Mega 2560. The experimental results in motionless position show that the filter can reduce the gravity acceleration. We have to set the initial value of q and R parameters in the estimation of position, speed, and acceleration using Kalman filter. Once R is decided, the change of q will determine K_k gain, and it will locate the poles of the observer that influence the stability and the estimation result. With $R=0.00005$ and $q=1$, the poles of Kalman filter are located in the unit circle so that the estimation is stable and appropriate with the data from the sensor and even cancel the noise.

Keywords: *Wii-Nunchuk, accelerometer, Kalman filter*

Copyright © 2017 Jurnal Rekayasa Elektrika. All right reserved

I. PENDAHULUAN

Interaksi antara manusia dan komputer/mesin tidak lagi terbatas pada antarmuka teks dan grafis dengan *Graphical User Interface* (GUI), namun semakin berkembang menjadi *Perceptual User Interface* (PUI) melalui pengenalan citra, suara, maupun gerakan. Interaksi melalui gerakan telah digunakan oleh beberapa produsen *video game* untuk membuat interaksi lebih nyata. Gerakan tangan dan tubuh dapat diketahui melalui kamera, misalnya yang diterapkan pada *game console* XBOX dengan *Kinect* dari *Microsoft*, atau menggunakan *joystick/controller*

yang mempunyai sensor gerak akselerometer tiga sumbu, pada *game console* *Wii* dengan *Wiimote* dan *Wiinunchuk controller* dari *Nintendo*. Meskipun *controller Wii* awalnya digunakan khusus untuk *game console* *Wii*, namun dengan fitur konektivitas *wireless bluetooth* pada *Wiimote* dan akselerometer pada *Wiimote* dan *WiiNunchuk*, *controller* ini dapat dikembangkan pada berbagai fungsi yang lebih luas [1].

Dua permasalahan utama dalam penggunaan akselerometer, pertama bagaimana sensor mengenali gerakan sebagai perubahan posisi karena data yang diperoleh sensor akselerometer hanya berupa data

percepatan dan kedua bagaimana mengatasi derau yang muncul yang dapat mempengaruhi hasil pembacaan.

Filter Kalman digunakan untuk mengatasi permasalahan tersebut karena selain dapat mereduksi derau, juga bekerja sebagai *state observer* yang berfungsi untuk mengestimasi *state* kecepatan dan perpindahan tanpa harus melakukan pengukuran secara langsung, namun cukup dari pengukuran *state* percepatan yang dapat diketahui melalui sensor.

Penelitian tentang estimasi posisi menggunakan akselerometer telah dilakukan diantaranya oleh Pratama dan Widyawan, yang memanfaatkan akselerometer pada *smartphone* untuk menentukan posisi dan mendeteksi langkah pejalan kaki dengan metode estimasi perpindahan melalui pendekatan pantulan langkah serta filter *lowpass* dan *highpass* untuk mengatasi deraunya, sehingga ini menggunakan dua algoritma secara terpisah [2]. Terdapat pula penelitian yang dilakukan oleh Prawira Negara dan Laksono yang juga memanfaatkan percepatan statis akselerometer pada *smartphone* yang digunakan untuk mengatur sudut pergerakan untuk robot beroda menggunakan logika *fuzzy* [3]. Namun penelitian [2] dan [3] tidak dapat mengestimasi kecepatan, perpindahan dan tidak memfilter data akselerometer padahal pembacaan sensor akselerometer cukup banyak menghasilkan derau.

Penelitian yang dilakukan oleh Liu dan Pang mengimplementasikan Filter Kalman untuk mengestimasi posisi robot lengan dengan akselerometer ADXL 202 pada mikrokontroler Motorola 68HC11F1 [4]. Sedangkan penelitian lain oleh Riyadi, Wahyudi dan Setiawan menggunakan akselerometer MMA 7260Q untuk mengestimasi kecepatan dan posisi dengan membandingkan performa antara filter eksponensial dan filter Kalman kemudian mengimplementasikannya ke mikrokontroler ATmega32 [5]. Meskipun telah mengimplementasikan Filter Kalman, namun pada penelitian [4] dan [5] pengaruh percepatan gravitasi akibat kemiringan sudut pembacaan sensor belum diperhitungkan padahal ini berpengaruh pada hasil pembacaan percepatan linier akselerometer ketika akselerometer dipegang tidak dalam kondisi lurus [6].

Penelitian yang dilakukan oleh Wahyudi, Susanto, Pramono dan Widada mengestimasi posisi melalui sensor akselerometer menggunakan Filter Kalman [7]. Terdapat pula penelitian oleh Singhal, Harid, dan Vishwakarma yang juga menggunakan Filter Kalman sebagai estimasi posisi dan kecepatan dengan melihat pengaruh perubahan variabel R dan q [8]. Namun pada penelitian [7] dan [8] masih berupa simulasi. Penelitian oleh Akbar, Wahyudi, dan Hidayatno menggunakan akselerometer ADXL 345 untuk mengestimasi posisi dan sudut menggunakan perpaduan sensor akselerometer dan giroskop yang sudah memperhitungkan pengaruh gravitasi dan diimplementasikan pada mikrokontroler ATmega 32 dengan *complementary* filter [9]. Namun pada penelitian ini tidak menggunakan Filter Kalman, padahal dari penelitian [5] disimpulkan bahwa hasil estimasi Filter Kalman menghasilkan nilai yang lebih baik. Selain

itu sensor yang digunakan [4], [5], [9] masih berupa perangkat langsung dan belum dikemas secara *compact* sehingga pengguna kurang dapat melakukan interaksi gerakan dengan nyaman.

Sehingga pada penelitian ini bertujuan untuk mengestimasi perpindahan posisi menggunakan Filter Kalman dengan perbaikan koreksi pengaruh percepatan gravitasi akibat sudut kemiringan saat memegang *joystick* akan diimplementasikan pada Arduino Mega 2560 yang dapat menjadi alternatif interaksi manusia dengan mesin berbasis pengenalan gerakan menggunakan *joystick WiiNunchuk* yang *compact* untuk kenyamanan pengguna.

II. PEMODELAN SISTEM DAN FILTER KALMAN

Sistem dimodelkan dalam ruang keadaan orde tiga dengan perpindahan s , kecepatan v , dan percepatan a , dipilih sebagai *state* yang hubungan ketiganya dinyatakan oleh,

$$\frac{ds}{dt} = v; \quad \frac{dv}{dt} = a; \quad \frac{da}{dt} = 0. \quad (1)$$

Persamaan (1) diubah ke ranah diskrit k dengan metoda transformasi bilinear dengan waktu cuplik T_s menjadi,

$$s_k = s_{k-1} + T_s v_k + \frac{1}{4} T_s^2 a_k + \frac{1}{4} T_s^2 a_{k-1}. \quad (2)$$

$$v_k = v_{k-1} + \frac{1}{2} T_s a_k + \frac{1}{2} T_s a_{k-1}. \quad (3)$$

$$a_k = a_{k-1} \quad (4)$$

Karena $a_k = a_{k-1}$, sehingga (2) dan (3) dapat disederhanakan menjadi (5) dan (6),

$$s_k = s_{k-1} + T_s v_k + \frac{1}{2} T_s^2 a_{k-1}. \quad (5)$$

$$v_k = v_{k-1} + T_s a_{k-1}. \quad (6)$$

$$a_k = a_{k-1} \quad (7)$$

Persamaan (5), (6), (7) dapat dinyatakan dalam bentuk ruang keadaan dengan vektor keadaan x yang terdiri dari perpindahan, kecepatan, percepatan, dan persamaan keluaran z ,

$$x_k = \begin{pmatrix} s_k \\ v_k \\ a_k \end{pmatrix}; \quad z_k = s_k$$

menjadi

$$x_k = \begin{pmatrix} 1 & T_s & \frac{1}{2} T_s^2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{pmatrix} x_{k-1}, \quad (8)$$

$$z_k = (1 \ 0 \ 0) x_k. \quad (9)$$

Nilai percepatan diperoleh dari pembacaan sensor akselerometer *WiiNunchuk* tiap waktu cuplik $T_s=1$ milidetik.

Filter Kalman bekerja sebagai estimator dari *state* yang nilainya tidak dapat diketahui melalui sensor, pada kasus ini adalah *state* kecepatan dan *state* perpindahan. Terdapat beberapa persamaan yang perlu diketahui dalam proses Filter Kalman [10]. *State* yang akan diestimasi menggunakan Filter Kalman diskrit, mempunyai persamaan ruang keadaan dan keluaran,

$$\begin{aligned} x_k &= Ax_{k-1} + Bu_{k-1} + w_{k-1}, \\ z_k &= Hx_k + v_k. \end{aligned} \tag{10}$$

Matriks *A* menghubungkan *state* waktu diskrit sekarang *k* dengan sebelumnya *k-1*. Matriks *B* menghubungkan input kontrol *u* dengan *state* *x*. Matriks *H* menghubungkan *state* *x* dengan pengukuran *z*, w_k dan v_k mewakili derau proses dan derau pengukuran yang saling bebas, berjenis *white noise* dengan probabilitas berdistribusi normal dengan kovariansi masing-masing *Q* dan *R*.

Didefinisikan beberapa variabel yaitu *state* pra estimasi \hat{x}_k^- , *state* pasca estimasi \hat{x}_k , galat pra estimasi \hat{e}_k^- , galat pasca estimasi \hat{e}_k . Dimana galat pra estimasi $\hat{e}_k^- = x_k - \hat{x}_k^-$ dan galat pasca estimasi $\hat{e}_k = x_k - \hat{x}_k$. Matriks kovariansi galat pra estimasi $P_k^- = [\hat{e}_k^- \hat{e}_k^{-T}]$, dan matriks kovariansi galat pasca estimasi $P_k = [\hat{e}_k \hat{e}_k^T]$, serta K_k adalah faktor pengali yang meminimumkan matriks kovarian galat pasca estimasi (P_k).

Persamaan Filter Kalman terdiri dari dua bagian, yaitu persamaan prediksi dan persamaan koreksi. Pada persamaan prediksi, variabel yang akan diprediksi adalah *state* pra estimasi \hat{x}_k^- dan matriks kovariansi galat pra estimasi P_k^- waktu cuplik saat ini, berdasarkan estimasi *state* dan matriks kovarian galat pasca estimasi pada waktu cuplik sebelumnya. Persamaan prediksi adalah,

$$\begin{aligned} \hat{x}_k^- &= A\hat{x}_{k-1} + Bu_{k-1}, \\ P_k^- &= AP_{k-1}A^T + Q. \end{aligned} \tag{11}$$

Pada persamaan koreksi, variabel yang dikoreksi adalah *state* pasca estimasi \hat{x}_k serta matriks kovarian pasca estimasi P_k , yang mana P_k dipengaruhi oleh faktor pengali K_k yang menyebabkan nilai P_k minimum. Persamaan koreksi adalah,

$$\begin{aligned} K_k &= P_k^- H^T (HP_k^- H^T + R)^{-1}, \\ \hat{x}_k &= \hat{x}_k^- + K_k (z_k - H\hat{x}_k^-), \\ P_k &= (1 - K_k H) P_k^-. \end{aligned} \tag{12}$$

Persamaan prediksi dan koreksi ini dihitung berurutan terus-menerus. Performa filter dapat diatur melalui penyesuaian parameter *Q* dan *R* yang berpengaruh pada nilai *gain* K_k . Bila kovariansi derau pengukuran *R* diatur mendekati nol, hasil pengukuran z_k lebih dipercaya dibandingkan hasil prediksi $H\hat{x}_k^-$; sebaliknya jika matriks kovariansi galat pra estimasi P_k^- mendekati nol yang terpengaruh dari kovariansi derau proses *Q* yang diatur mendekati nol, maka hasil prediksi $H\hat{x}_k^-$ lebih dipercaya dibandingkan hasil pengukuran z_k .

III. METODE PERANCANGAN DAN IMPEMENTASI

WiiNunchuk mempunyai akselerometer tiga sumbu LIS3L02AL dengan rentang maksimum ± 2 g, dari *ST Microelectronics* dengan 10-bit ADC yang berkomunikasi menggunakan protokol I²C sebagai berikut (Gambar 1) [11][13]:

- Inisialisasi perintah agar *wiiNunchuk* siap menerima dan mengirim data dengan cara mengirim data sebagai berikut:

I ² C start	Write 0x04	Slave ACK	Write 0x40	Slave ACK	Write 0x00	I ² C stop
------------------------	------------	-----------	------------	-----------	------------	-----------------------

- Mengirim perintah agar data sensor disimpan ke dalam 6-byte register di dalam *WiiNunchuk*.

I ² C start	Write 0x04	Slave ACK	Write 0x00	I ² C stop
------------------------	------------	-----------	------------	-----------------------

- Mengirimkan perintah *slave* ID 0xA5 untuk mengambil data dari *WiiNunchuk* dan menerima 6-byte data dengan setiap satu data telah diterima, kirimkan ACK ke *WiiNunchuk* sebagai tanda bahwa data telah diterima oleh *master* (Arduino).

I ² C start	Write 0xA5	Slave ACK	Byte1	Master ACK	Byte2	Master ACK	Byte3
------------------------	------------	-----------	-------	------------	-------	------------	-------

Master ACK	Byte4	Master ACK	Byte5	Master ACK	Byte6	Master	No ACK	I ² C stop
------------	-------	------------	-------	------------	-------	--------	--------	-----------------------

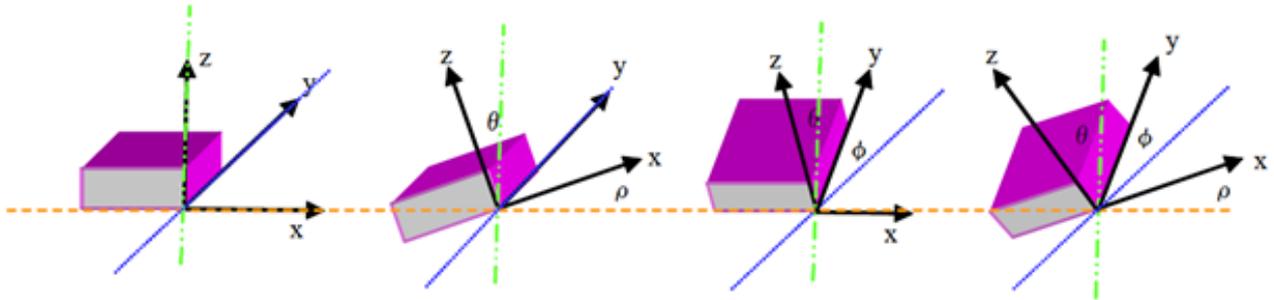
6-byte data yang diterima mikrokontroler adalah sebagai berikut:

byte0: data analog stick X							
byte1: data analog stick Y							
byte2: data akselerometer X (bit 9:2 dari resolusi 10 bit)							
byte3: data akselerometer Y (bit 9:2 dari resolusi 10 bit)							
byte4: data akselerometer Z (bit 9:2 dari resolusi 10 bit)							
Accel.z bit 1	Accel.z bit 0	Accel.y bit 1	Accel.y bit 0	Accel.x bit 1	Accel.x bit 0	C	Z

Data yang dibutuhkan pada penelitian ini adalah data 10 bit akselerometer yang diperoleh dari, untuk sumbu-x gabungan *byte* 2 dengan bit 2 dan 3 *byte*5, sumbu-y *byte* 3 dengan bit 4 dan 5 *byte* 5, sumbu-z *byte* 4 dengan bit 6 dan 7 *byte* 5. Karena menggunakan protokol I²C, dibutuhkan rangkaian resistor *pull-up* pada kaki SCL dan SDA pada mikrokontroler.

Setelah data *digital* 10 bit ketiga sumbu akselerometer diketahui, dilakukan kalibrasi untuk mencari posisi titik nol ketiga sumbu dan mengubahnya ke skala gravitasi *g* [12].

Supaya nilai percepatan yang diperoleh hanya percepatan linier, tidak terpengaruh percepatan gravitasi akibat posisi sensor yang miring, nilai percepatan (*g*) masing-masing sumbu dikurangkan dengan pengaruh sudut akibat percepatan gravitasi [14] yang merupakan nilai sinus dari sudut ρ , ϕ , θ [8]. Dengan ρ , ϕ , θ masing-masing adalah [13]:



Gambar 1. Orientasi sudut tiga sumbu [13]

$$\rho = \tan^{-1} \left(\frac{A_x}{\sqrt{A_y^2 + A_z^2}} \right); \tag{13}$$

$$\phi = \tan^{-1} \left(\frac{A_y}{\sqrt{A_x^2 + A_z^2}} \right);$$

$$\theta = \tan^{-1} \left(\frac{\sqrt{A_x^2 + A_y^2}}{A_z} \right)$$

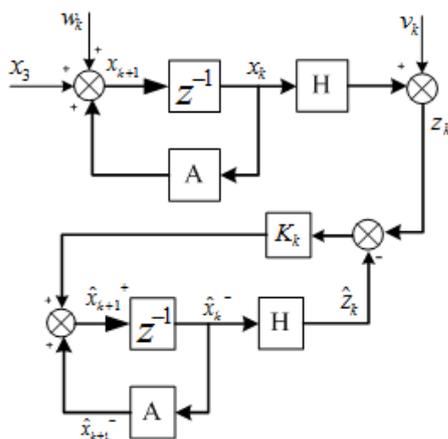
Setelah diperoleh percepatan linier dalam g masing-masing sumbu, nilai tersebut $\times 980 \text{ cm/s}^2$ sehingga akan diperoleh nilai percepatan linier dalam cm/s^2 . Nilai inilah yang akan diolah Filter Kalman dengan blok diagram sebagaimana diperlihatkan pada Gambar 2 sebagai *state* ketiga (x_3).

Proses pembacaan, pengolahan sensor, dan Filter Kalman diimplementasikan pada Arduino Mega 2560 dengan waktu cuplik 1 milidetik dengan diagram alir terlihat pada Gambar 3.

IV. HASIL DAN PEMBAHASAN

Implementasi perangkat keras Arduino Mega 2560, *WiiNunchuk* dan rangkaian resistor *pull-up* terlihat pada Gambar 4. Hasil pengolahan data dan Filter Kalman dari Arduino dikirimkan tiap 5 milidetik ke MATLAB Simulink agar dapat dilihat dan disimpan untuk analisis.

Pengujian dilakukan pada salah satu sumbu yaitu sumbu-x dengan cara menggerakkan *joystick* ke kanan dan kiri berulang kali dengan jarak dan kecepatan yang

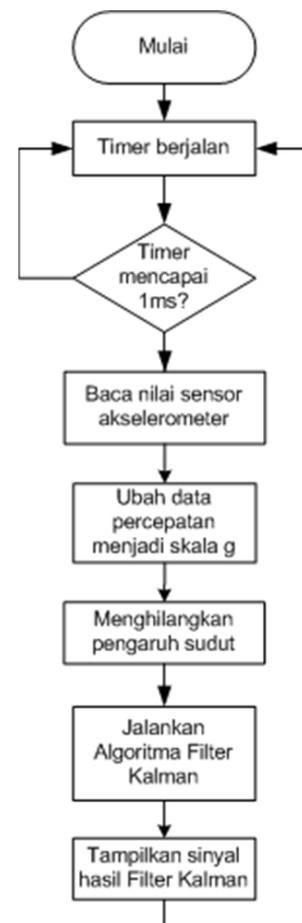


Gambar 2. Blok diagram filter Kalman

bervariasi. Pertama akan dilakukan pengujian terhadap implementasi penghilangan pengaruh sudut akibat gravitasi ditunjukkan pada Gambar 5.

Dari Gambar 5 tampak bahwa nilai percepatan asli pada saat akselerometer dalam kondisi diam yang seharusnya bernilai nol, masih mempunyai nilai akibat pengaruh percepatan gravitasi. Hal ini dapat mengakibatkan akumulasi galat pada estimasi kecepatan dan jarak sebagaimana terlihat pada Gambar 6 dan Gambar 7. Hasil perbaikan yang diimplementasikan dengan menghilangkan pengaruh sudut (grafik biru) menghasilkan nilai mendekati nol saat kondisi diam.

Kemudian dilakukan pengujian estimasi percepatan dengan Filter Kalman dan tanpa Filter Kalman, yang mana keduanya sudah dihilangkan pengaruh sudutnya, dengan



Gambar 3. Diagram alir implementasi sistem pada Arduino

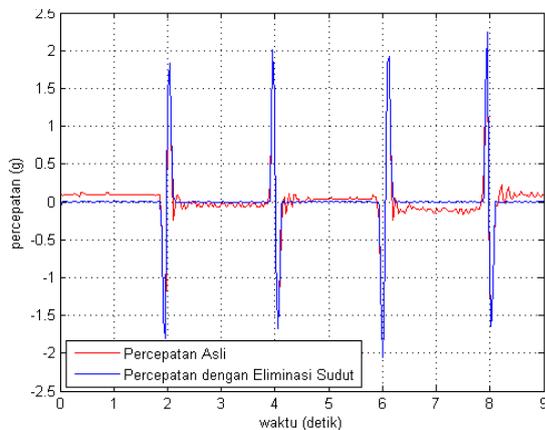


Gambar 4. Implementasi perangkat keras

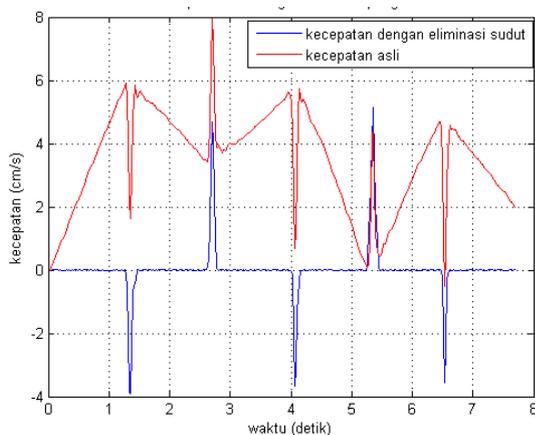
parameter Filter Kalman nilai q yang diubah-ubah dan nilai R dibuat tetap dengan $R=0,00005$, yang diperoleh dari *acceleration noise density* akselerometer LIS3L02AL.

Untuk $q=0,00001$, perbandingan percepatan, kecepatan, dan perpindahan hasil estimasi dengan percepatan, kecepatan, dan perpindahan asli dapat dilihat pada Gambar 8–10. Tampak bahwa hasil estimasi lambat dalam mengikuti perubahan percepatan dan muncul *overshoot*.

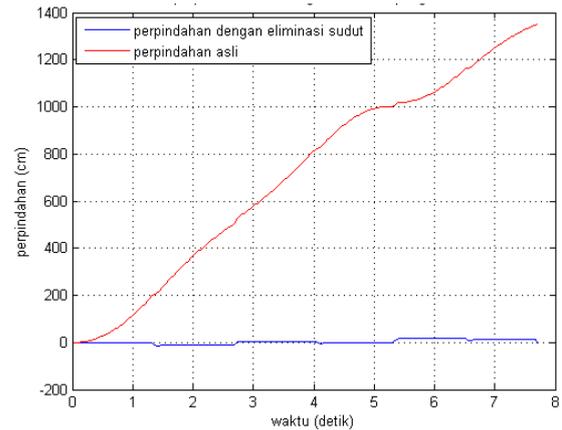
Untuk $q=1$, perbandingan percepatan, kecepatan dan perpindahan hasil estimasi dengan percepatan, kecepatan



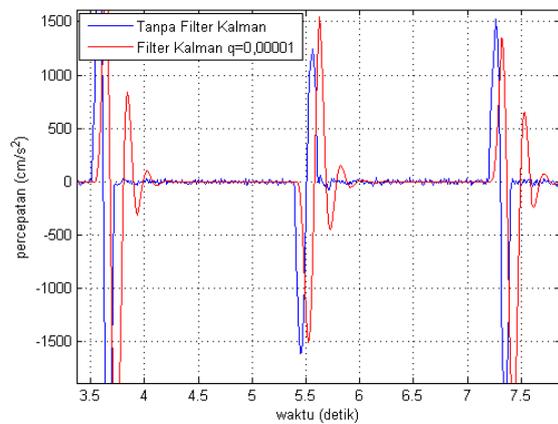
Gambar 5. Perbandingan percepatan asli dan eliminasi pengaruh sudut



Gambar 6. Perbandingan kecepatan asli dan eliminasi pengaruh sudut



Gambar 7. Perbandingan perpindahan asli dan eliminasi pengaruh sudut



Gambar 8. Percepatan dengan Filter Kalman $q=0,00001$

dan perpindahan asli dapat dilihat pada Gambar 11-13. Tampak bahwa hasil estimasi mampu mengatasi derau dengan respon yang cepat.

Untuk $q=1000$, terlihat pada Gambar 14 bahwa hasil estimasi percepatan semakin membesar, berosilasi, dan tidak stabil. Hal tersebut terjadi karena pada $q=1000$ gain

$$K_k = \begin{pmatrix} 1 \\ 20000,2 \\ 2000000,65 \end{pmatrix},$$

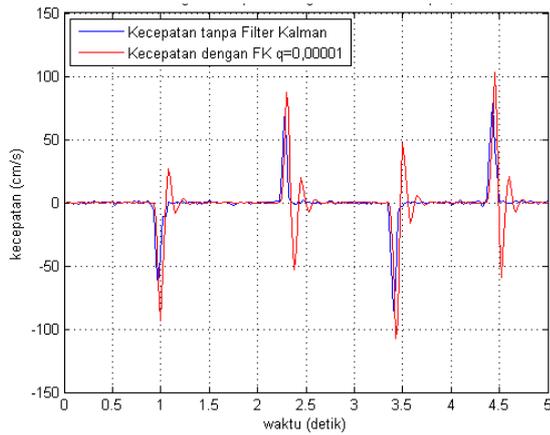
sehingga kutub-kutub *observer* (nilai eigen dari $(A-K_kH)$) berada pada $z_{1,2}=0,857 \pm 1,666i$ dan $z_3=0,284$. Terdapat kutub yang terletak di luar lingkaran satuan ($z_{1,2}$), yang membuat hasil estimasi sistem tidak stabil.

Sedangkan pada $q=0,00001$ gain

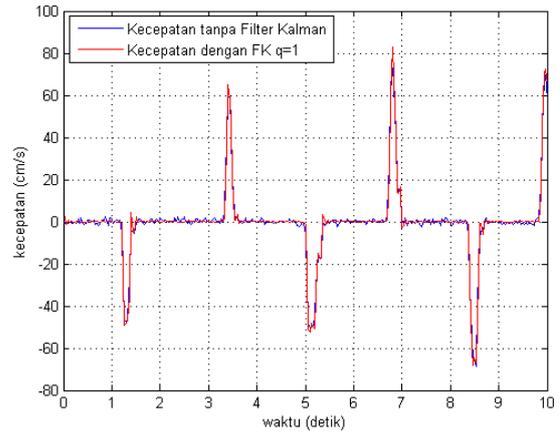
$$K_k = \begin{pmatrix} 0,11 \\ 7,97 \\ 354,17 \end{pmatrix},$$

sehingga kutub-kutub *observer* berada pada $z_{1,2}=0,978 \pm 0,0693i$ dan $z_3=0,284$. Terdapat kutub ($z_{1,2}$) yang terletak dekat dengan batas kestabilan (lingkaran satuan), yang membuat hasil estimasi sistem berosilasi dan mempunyai *overshoot*.

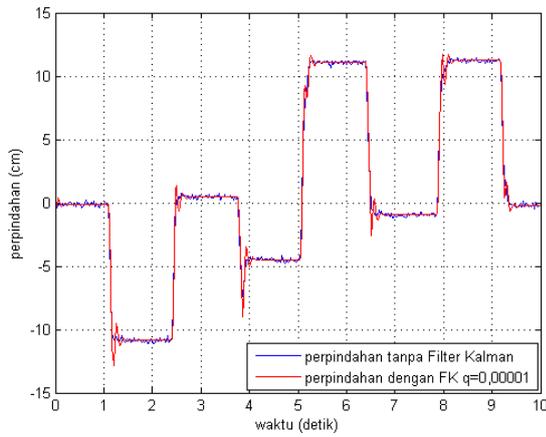
Sedangkan pada $q=1$ gain



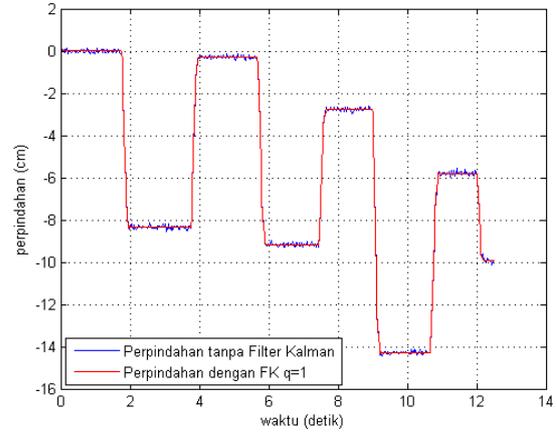
Gambar 9. Kecepatan dengan Filter Kalman q=0,00001



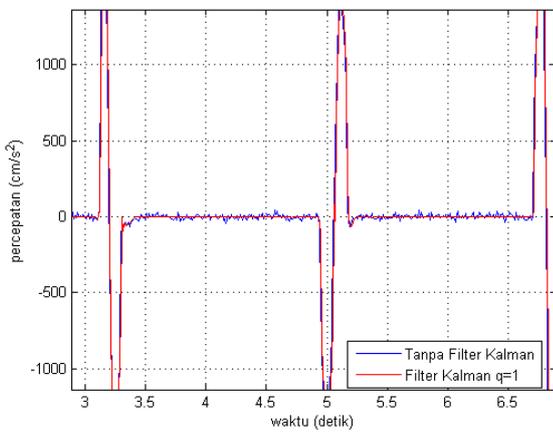
Gambar 12. Kecepatan dengan Filter Kalman dengan q=1



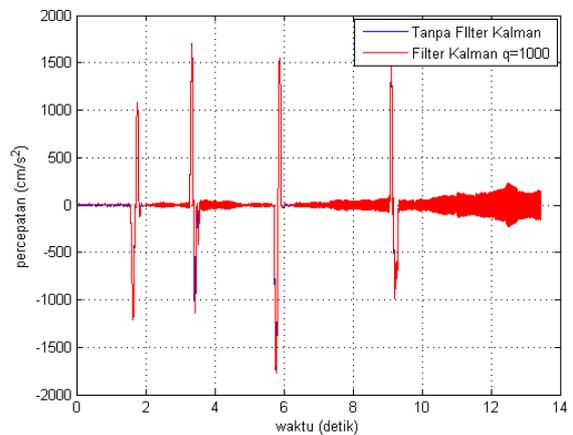
Gambar 10. Perpindahan dengan Filter Kalman q=0,00001



Gambar 13. Perpindahan dengan Filter Kalman dengan q=1



Gambar 11. Percepatan dengan Filter Kalman dengan q=1



Gambar 14. Percepatan dengan Filter Kalman dengan q=1000

$$K_k = \begin{pmatrix} 0,68 \\ 469,69 \\ 194152,84 \end{pmatrix}$$

sehingga kutub-kutub *observer* berada pada $z_{1,2} = 0,871 \pm 0,664i$ dan $z_3 = 0,576$. Kutub-kutub ini terletak di dalam lingkaran satuan sehingga estimasi sistem stabil serta hasil estimasi masih bersih dari derau.

V. KESIMPULAN

Berdasarkan pengujian dapat disimpulkan bahwa implementasi Filter Kalman telah berhasil. Dapat dilihat pula bahwa saat posisi diam tidak ada gerakan, namun dengan pemegangan *joystick* yang miring, sensor memberikan nilai percepatan gravitasi statis yang dapat menjadi akumulasi galat pada posisi. Nilai percepatan gravitasi statis ini juga telah berhasil dihilangkan dengan menghilangkan pengaruh sudut.

Pada implementasi Filter Kalman, perlu diperhatikan penentuan parameter q dan R . Saat R tetap, perubahan

nilai q akan menentukan nilai K_k yang menentukan lokasi kutub-kutub dari *observer* yang berpengaruh pada kestabilan dan hasil estimasi. Pemilihan nilai $R=0,00005$ diperoleh dari *acceleration noise density* akselerometer *WiiNunchuk* LIS3L02AL, sedangkan nilai q diperoleh melalui penalaan dengan nilai $q=1$ merupakan hasil yang terbaik karena hasil estimasi masih mengikuti sensor namun bersih dari derau serta dihasilkan sistem yang stabil karena semua kutub-kutub *observer* Kalman berada pada lingkaran satuan.

REFERENSI

- [1] Y Yand dan L li, "Turn a Nintendo Wiimote Into a Handheld Computer Mouse", *IEEE Potentials*, vol. 30 Issue 1 hal.12-16, Jan. 2011.
- [2] A. R. Pratama dan Widyawan, "Pedestrian Dead Reckoning Pada Ponsel Cerdas Sebagai Sistem Penentuan Posisi Dalam Ruang", *JNTEI*, vol. 2, No. 3, hal. 20-25, Agt. 2013
- [3] M. A. Prawira Negara dan D. S. Laksono, "Perancangan Kendali Robot pada Smartphone Menggunakan Sensor Akselerometer Berbasis Metode Fuzzy Logic", *Jurnal Rekayasa Elektrika*, vol. 3 No. 2, hal. 76-81, Agt. 2017.
- [4] H. H. S. Liu dan G. K. H. Pang, "Akselerometer for Mobile Robot Positioning", *IEEE Transactions on Industry Applications*, vol. 37, No. 3, hal 812-819, Mei 2001.
- [5] M. Riyadi, Wahyudi, dan I. Setiawan, "Pendeteksi Posisi Menggunakan Sensor Akselerometer MMA7260Q Berbasis Mikrokontroler Atmega32", *TRANSMISI*, vol. 12, No. 2, hal 76-81, 2010.
- [6] J. R. Nistler dan M. F. Selekwa, "Gravity Compensation in Akselerometer Measurements for Robot Navigation on Inclined Surface", *Procedia Computer Science*, vol. 6, hal413-418, 2011.
- [7] Wahyudi, A. Susanto, S. Pramono H, dan W. Widada, "Simulasi Filter Kalman Untuk Estimasi Posisi dengan Menggunakan Sensor Akselerometer", *Jurnal Techno Science*, vol. 3, No. 1, hal 350-359, Mei 2009.
- [8] T. Singhal, A. Harit, dan D N Vishwakarma, "Kalman Filter Implementation on an Akselerometer Sensor Data for Three State Estimation of a Dynamic System", *IJRET*, vol. 1 No. 6, hal 330-334 2012.
- [9] M. A. Akbar, Wahyudi, dan A. Hidayatno, "Perancangan Perangkat Lunak Pendeteksi Sudut dan Posisi Menggunakan Mikrokontroler Atmega32", *TRANSMISI*, vol. 16, No. 1, hal 49-53, 2014.
- [10] G. Welch dan G. Bishop, *An Introduction to The Kalman Filter*, UNC-Chapel Hill, TR95-041, Jul. 2006.
- [11] Anonym, ZX-Nunchuk Wii-Nunchuk Interface Board.
- [12] K. Tuck, *Impelementing Auto-Zero Calibration Technique for Akselerometers*, AN3447 Freescale Semiconductor Application Note, Rev 0, Mar. 2007.
- [13] K. Tuck, *Tilt Sensing Using Linear Akselerometers*, AN3461 Freescale Semiconductor Application Note, Rev 2, Jun. 2007.
- [14] C. Fischer, P. T. Sukumar, M. Hazas, "Tutorial: Implementing a Pedestrian Tracker Using Inertial Sensors", *IEEE Pervasive Computing*, vol. 12 Issue 2, hal 17-27, Jan 2012.

Penerbit:

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Syiah Kuala

Jl. Tgk. Syech Abdurrauf No. 7, Banda Aceh 23111

website: <http://jurnal.unsyiah.ac.id/JRE>

email: rekayasa.elektrika@unsyiah.net

Telp/Fax: (0651) 7554336

