

Estimasi sudut orientasi rigid body dengan menggunakan sensor Inertial Measurement Unit (IMU) dan Magnetometer

Briyan Yoga Sandi*, Freddy Kurniawan, Lasmadi

Department of Electrical Engineering, Sekolah Tinggi Teknologi Adisutjipto

Email: briyansandi2@gmail.com

Abstract. *One of the instruments that can be used today for navigation is the IMU (Inertial Measurement Unit) sensor which consists of an accelerometer sensor, gyroscope, and an additional magnetometer. The three sensors are useful for determining the orientation angle. However, the presence of high enough noise in the three-output data can cause the orientation angle to be inaccurate. This study aims to determine the roll, pitch and yaw orientation more accurately using a Kalman filter. This filter is used because it has a lightweight computation and can reduce noise significantly. The results of this study show that the orientation angle calculated using the Kalman filter is close to the actual orientation angle with an accuracy of 99.2% for roll, 99.5% for pitch, and 98.6% for yaw.*

Keywords: *Inertial Measurement Unit, orientation, Kalman filter*

1. Pendahuluan

Navigasi adalah proses memantau dan mengendalikan pergerakan dari seseorang atau alat transportasi (mobil atau kapal atau pesawat) dari satu tempat ke tempat lainnya. IMU merupakan sensor yang mampu mendeteksi pergerakan pada sumbu x, y, dan z yang akan memberikan kemudahan serta kenyamanan bagi penggunaannya. IMU ini banyak digunakan dalam sistem navigasi yang kemudian dikombinasikan dengan GPS (*Global Positioning System*) [1]-[14]. Sensor IMU (*Inertial Measurement Unit*) memiliki suatu kelemahan, seperti pada sensor akselerometer yang mana tidak bisa mendeteksi arah hadap suatu benda, maka perlu dikombinasi dengan sensor magnetometer dan sensor giroskop untuk mengetahui kecepatan besar medan magnet bumi. Sensor pada IMU (*Inertial Measurement Unit*) memiliki derau pada sistemnya di mana dapat mengganggu hasil pengukuran dari sensor pada IMU. Jika gangguan ini dibiarkan maka akan terjadi kesalahan hasil pengukuran yang akan berpengaruh pada salahnya informasi navigasi.

Pada penelitian ini, dilakukan pengolahan data antara sensor akselerometer, giroskop dan sensor magnetometer yang bertujuan untuk dapat mengetahui *attitude* benda, berupa sudut orientasi *pitch*, *roll* dan *yaw* serta mengurangi derau dari sensor akselerometer, giroskop dan sensor magnetometer menggunakan *Kalman Filter*

2. Metodologi Penelitian

Penelitian ini mengolah data dari data keluaran aplikasi pada *smartphone* yang berupa data dari sensor akselerometer, magnetometer, dan giroskop.



2.1. Kalman Filter

Kalman Filter merupakan salah satu algoritma yang mampu mengestimasi kondisi selanjutnya. Kalman Filter memperkirakan hasil berikutnya dengan berdasarkan data yang sudah ada sebelumnya. Semisal jika diketahui suatu posisi dengan kecepatan dan waktu tertentu, maka posisi selanjutnya dapat diperkirakan dengan Kalman Filter ini. Keunggulan Kalman Filter selain untuk mengestimasi keadaan selanjutnya adalah mampu mengurangi derau data yang diestimasi. Sehingga dengan keunggulan tersebut, Kalman Filter juga dipakai sebagai filter digital. Dalam hal ini dapat dilihat pada persamaan (1).

$$\mathbf{x}_k = \mathbf{C}\mathbf{x}_{k-1} + \mathbf{D}\mathbf{u}_k + \mathbf{w}_k \quad (1)$$

Keterangan,

\mathbf{x}_k = proses prediksi pada waktu saat ini.

\mathbf{x}_{k-1} = proses prediksi pada waktu sebelumnya.

\mathbf{E} = matriks transisi.

\mathbf{F} = matriks kontrol masukan.

\mathbf{u}_k = data masukan berdasarkan waktu ke k.

\mathbf{w}_k = derau proses.

dimana Dengan model pengukuran z merupakan bilangan *real* yang mana

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \quad (2)$$

Keterangan,

\mathbf{z}_k = hasil pengukuran.

\mathbf{H} = matriks pengukuran.

\mathbf{x}_k = hasil proses prediksi.

\mathbf{v}_k = derau pengukuran

Pada persamaan (2) selanjutnya dilakukan observasi dengan menggunakan pemodelan pengukuran yang memetakan $state \times$ ke keluaran z yang dapat di observasi. Model pemetaan keluaran z ditunjukkan dalam persamaan (3).

$$\mathbf{z}_k = \mathbf{H}\hat{\mathbf{x}}_k^- + \mathbf{v}_k \quad (3)$$

keterangan :

\mathbf{z}_k : data hasil pengukuran

\mathbf{H} : model matriks pengukuran

$\hat{\mathbf{x}}_k^-$: hasil prediksi perhitungan *state*

\mathbf{v}_k : derau pengukuran

Derau proses (\mathbf{w}_k) dan derau pengukuran (\mathbf{v}_k) merupakan derau yang saling bebas. Nilai estimasi *state* $\hat{\mathbf{x}}_k$ pada *Kalman Filter* ditentukan dari estimasi *posteriori* $\hat{\mathbf{x}}_k^-$ serta selisih antara pengukuran sebenarnya \mathbf{z}_k dan estimasi pengukuran $\mathbf{H}\hat{\mathbf{x}}_k^-$ seperti pada persamaan (4)

$$\begin{aligned} \hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_k^-) \\ \hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{H}\hat{\mathbf{x}}_k^- + \mathbf{v}_k - \mathbf{H}\hat{\mathbf{x}}_k^-) \end{aligned} \quad (4)$$

keterangan :

$\hat{\mathbf{x}}_k$: *state* dari sistem pada waktu k

$\hat{\mathbf{x}}_k^-$: hasil prediksi perhitungan *state*

\mathbf{K}_k : Kalman gain

\mathbf{z}_k : data hasil pengukuran

\mathbf{v}_k : derau pengukuran

\mathbf{H} : model matriks pengukuran

Selisih nilai antara pengukuran sebenarnya \mathbf{z}_k dan estimasi pengukuran disebut sebagai *residual* atau pengukuran *innovation*. Jika nilai *residual* adalah nol, maka hal itu menunjukkan bahwa hasil estimasi sama dengan hasil pengukuran sensor. Semakin tinggi R, maka semakin besar derau *measurement*, yaitu derau perhitungan sudut dari sensor akselerometer dan magnetometer.

Nilai K_k adalah faktor *gain* pada *Kalman Filter*. Pada *Kalman Filter* dipilih nilai K_k sehingga estimasi *posteriori* adalah optimal atau mempunyai *error* yang minimum. Adapun persamaan faktor *gain* pada *Kalman Filter* ditunjukkan dalam persamaan (5).

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (5)$$

keterangan :

- K_k : nilai faktor *gain* pada *Kalan Filter* (*Kalman Gain*)
- P_k^- : prediksi kovarian *error*
- H^T : matriks transpos dari model matriks pengukuran
- R : kovarian derau pengukuran
- H : model matriks pengukuran

Nilai P_k^- minimum diperoleh jika nilai K_k dapat menyediakan estimasi yang mempunyai kovarian minimum. Semakin tinggi nilai Q , maka semakin besar derau proses, yaitu derau perhitungan sudut dari sensor giroskop di mana perhitungan sensor kurang tepat kebenarannya, kemudian untuk mendapatkan P_k^- minimum ditunjukkan pada persamaan (6)

$$P_k^- = A P_{k-1}^- A^T + Q \quad (6)$$

keterangan :

- P_k^- : prediksi kovarian *error*
- A : model matriks transisi
- P_{k-1}^- : prediksi kovarian *error* pada waktu $k-1$
- A^T : matriks transpos dari model matriks transisi
- Q : kovarian derau proses

Kemudian dilakukan *update* kovarian *error* yang berfungsi untuk mengetahui apakah masih terdapat derau pada *Kalman Filter*. Persamaan *update* kovarian *error* ditunjukkan dalam persamaan (7)

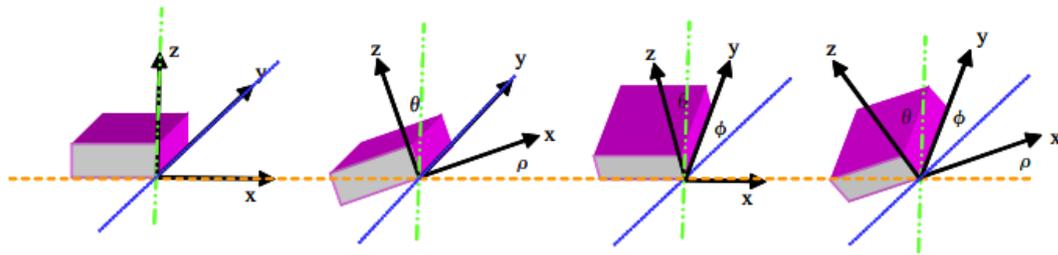
$$P_k = (I - K_k H) P_k^- \quad (7)$$

keterangan :

- P_k : *update* kovarian *error*
- I : matriks identitas
- K_k : nilai faktor *gain* pada *Kalan Filter* (*Kalman Gain*)
- H : model matriks pengukuran
- P_k^- : prediksi kovarian *error*

Proses tahapan kovarian *error* dilakukan secara terus menerus hingga mencapai waktu yang telah ditentukan. Sehingga mempunyai hasil *state* \hat{X}_k mendekati dengan hasil data yang digunakan sebagai pembanding.

2.2. Mencari Besar Sudut Sikap menggunakan Data Akselerometer



Gambar 1. Arah dari 3 sumbu (Tuck, 2007)

Untuk menentukan sudut akselerometer dalam tiga dimensi *pitch*, *roll* dan *yaw* dirasikan menggunakan ketiganya output dari akselerometer. *Pitch* (ρ) didefinisikan sebagai sudut dari sumbu X relatif ke tanah. *Roll* (ϕ) didefinisikan sebagai sudut dari sumbu Y relatif ke tanah. (Tuck, 2007).

Rumus yang didapat berupa :

$$\text{Pitch } (\rho) = \arctan\left(\frac{a_x}{\sqrt{a_y^2 + a_z^2}}\right) \quad (8)$$

$$\text{Roll } (\phi) = \arctan\left(\frac{a_y}{\sqrt{a_x^2 + a_z^2}}\right) \quad (9)$$

Bidang horizontal lokal benda didefinisikan sebagai bidang vector gravitasi bumi, jika arah benda berada di bidang horizontal lokal, maka sikap *roll* dan sikap *pitch* akan menjadi nol dan *heading*nya dirumuskan pada persamaan (10)

$$\text{Yaw } (\theta) = \arctan\left(\frac{y_H}{x_H}\right) \quad (10)$$

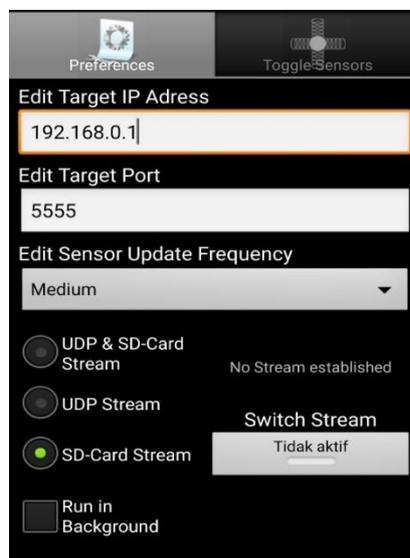
2.3. Pengambilan Data

Pengambilan data dilakukan dengan merekam data akselerometer, giroskop dan magnetometer dengan sensor K6DS3TR menggunakan aplikasi *Sensortream* IMU+GPS dengan waktu cuplik 0.06 detik, sehingga mempunyai nilai *sampling rate* 16-17 data per detik. Data yang digunakan untuk perhitungan orientasi *pitch* dan *roll* menggunakan sumbu x, y dan z dari data akselerometer dan data dari giroskop, sedangkan orientasi *yaw* (*heading*) menggunakan sumbu x, y dan z dari magnetometer dimana pada aplikasi *sensorstream* IMU+GPS berupa data dari “Magnetic Field”. Pengambilan data terlebih dahulu dilakukan dengan membuka aplikasi *Sensortream* IMU+GPS pada *smartphone*, langkah selanjutnya dengan mencentang pilihan “*Orientation*” dan pilihan “*Include User-Checked Sensor Data in Stream*” yang mana pilihan “*Accelerometer, Gyroscope, dan Magnetic Field*” sudah terplilih atau tercentang terlebih dahulu secara otomatis pada aplikasi *Sensortream* IMU+GPS yang ditunjukkan pada gambar 2, “*Include User-Checked Sensor Data in Stream*” berfungsi untuk menyimpan data yang sudah dicentang.



Gambar 2. Tampilan pilihan sensor

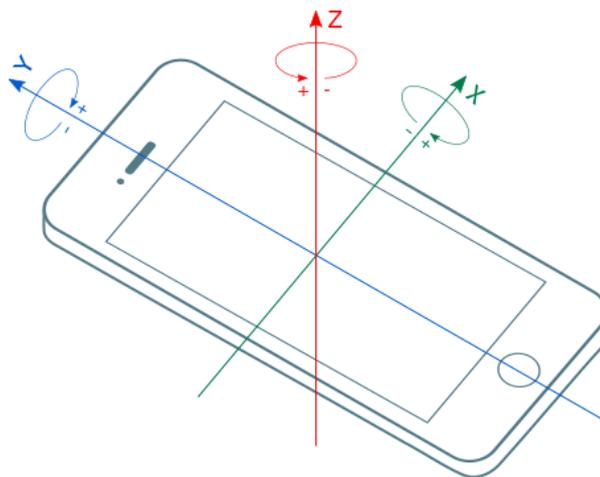
Untuk mengatur kecepatan frekuensi saat perekaman pengambilan data, dilihat pada pilihan “*Edit Sensor Update Frequency*” dengan berbagai pilihan seperti *slow*, *medium*, *fast*, dan *fastest*. Pada percobaan kali ini pengambilan data menggunakan pilihan frekuensi *medium* dan hasil pengambilan data disimpan pada penyimpanan *smartphone* dengan mencentang pilihan “*SD-Card Stream*” yang ditunjukkan pada gambar 3. Kemudian memilih pilihan “*Switch Stream*” untuk memulai perekaman dan kemudian pilih pilihan “*Switch Stream*” kembali untuk mematikannya kembali.



Gambar 3. Tampilan Pilihan perekaman data

2.4. Konversi Sumbu Koordinat

Konversi sumbu koordinat dilakukan untuk mengubah sumbu koordinat body (East North Up) pada saat smartphone diletakkan dengan bagian atas menghadap ke utara. Karena sensor K6DS3TR pada smartphone menggunakan koordinat body dengan sumbu y menghadap ke utara seperti ditunjukkan pada gambar 4, sedangkan algoritma yang digunakan pada perhitungan sudut orientasi yaw menggunakan sumbu koordinat body dengan sumbu x menghadap ke utara. Sehingga untuk mengolah data keluaran sensor perlu diubah pada sumbunya.



Gambar 4. Frame koordinat body (East North Up)

Berdasarkan gambar 4 merupakan arah sumbu koordinat benda atau body pada sensor K6DS3TR, yang mana dalam mengubah koordinat ini dengan cara menukar posisi data sumbu x dan data y pada koordinat body atau East North Up yang ditunjukkan pada gambar 4. Pada data sumbu z koordinat body perlu diubah menjadi data sumbu minus z koordinat body dengan cara membalikkan atau invers, sehingga jika data sumbu z koordinat body bernilai positif akan bernilai negatif pada data sumbu saat pemodelan algoritma

2.5. Pemodelan Program Orientasi

Hasil data dari sensor akselerometer, giroskop dan magnetometer kemudian diolah dengan memodelkan menggunakan rumus orientasi *pitch*, *roll*, *yaw* yang mana akan dimasukkan pada aplikasi matlab. Kode Program Orientasi *Roll* dan *Pitch* menggunakan data sumbu x,y,z dari sensor akselerometer dan sensor giroskop yang kemudian menggunakan rumus pada persamaan (11) dan (12).

$$Roll = \arctan \frac{acc_y}{\sqrt{acc_x^2 + acc_z^2}} \quad (11)$$

$$Pitch = \arctan \frac{acc_x}{\sqrt{acc_y^2 + acc_z^2}} \quad (12)$$

Orientasi *yaw* menggunakan data sumbu x,y,z dari sensor magnetometer pada persamaan (13).

$$Yaw = -\arctan 2 \frac{mag_y}{mag_x} \quad (13)$$

Dari ketiga persamaan tersebut menghasilkan pergerakan sudut dalam bentuk radian, sedangkan hasil yang diperlukan dalam bentuk derajat. Sehingga perlu mengubah dari bentuk radian ke derajat, di mana satu radian sama dengan $57,3^\circ / (180/\pi)$. Maka jika menginginkan hasil derajat dikalikan dengan $57,3^\circ / (180/\pi)$.

2.6. Pemodelan *Kalman Filter*

Sensor akselerometer, giroskop dan magnetometer masing-masing mempunyai 3 sumbu, yaitu sumbu x, y, z, kemudian dilambangkan sebagai ω_x , ω_y , dan ω_z serta dapat ditulis sebagai persamaan vektor pada persamaan (14).

$$\omega_i = [\omega_x \ \omega_y \ \omega_z]^T \quad (14)$$

Dalam perancangan *state space* pada penelitian kali ini, persamaan vektor x ditentukan pada persamaan (15).

$$x_k = [\theta \ \dot{\theta} \ \phi \ \dot{\phi} \ \varphi \ \dot{\varphi}]^T \quad (15)$$

dimana θ , ϕ , dan φ merupakan besar sudut sikap benda pada sumbu x, sumbu y, dan sumbu z, sedangkan $\dot{\theta}$, $\dot{\phi}$, dan $\dot{\varphi}$ merupakan kecepatan sudut pada sumbu x, sumbu y, dan sumbu z. Pada penelitian kali ini kecepatan diasumsikan pada titik [0,0], dan sudut diasumsikan tidak tetap dikarenakan disesuaikan dengan sensor yang dipakai yaitu sensor K6DS3TR.

Kalman Filter mempunyai dua proses utama, yaitu prediksi dan koreksi. Pada proses prediksi, *Kalman Filter* terdapat dua tahapan yaitu prediksi *state* dan prediksi kovarian *error*. Sedangkan pada proses koreksi, terdapat tiga tahapan. Tahapan yang pertama pada proses koreksi yaitu menghitung kalman gain, selanjutnya tahap yang kedua dilakukan *update* estimasi dengan pengukuran z_k dan tahapan yang ketiga yaitu *update* kovarian *error*.

Proses pertama pada *Kalman Filter* yaitu menentukan prediksi *state* awal yang ditunjukkan pada persamaan (16).

$$\hat{x}_k^- = E\hat{x}_{k-1} + Fu_k \quad (16)$$

di mana E dan F merupakan model dari matriks transisi, \hat{x}_k^- adalah prediksi *state* dari sistem pada waktu k, \hat{x}_{k-1} adalah *state* dari sistem pada waktu k-1, dan u_k merupakan masukan data dari sensor pada indeks waktu k. Model matriks E dan F didefinisikan dengan persamaan (17) dan (18)

$$E = \begin{bmatrix} 1 & dt \\ 0 & 0 \end{bmatrix} \quad (17)$$

$$F = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (18)$$

nilai dt merupakan selisih waktu dalam satuan detik. Apabila matriks E dan F disimulasikan dalam program, maka akan memiliki model matriks E 6x6 dan matriks F 6x3 dimana matriks tersebut masing-masing perkalian pada sumbu x, sumbu y, dan sumbu z dengan data masukan pada *gyroscope* sehingga matriks E dan F menjadi persamaan berikut

$$E = \begin{bmatrix} 1 & dt & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & dt & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & dt \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (19)$$

$$F = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (20)$$

Setelah menentukan prediksi *state* awal, kemudian menentukan prediksi kovarian *error* yang ditunjukkan pada persamaan (21)

$$P_k^- = AP_{k-1}A^T + Q \quad (21)$$

dimana P_k^- merupakan prediksi kovarian *error* pada waktu k, E merupakan model dari matriks transisi, P_{k-1} merupakan prediksi kovarian *error* pada waktu k-1 dan Q adalah *noise* kovarian *error*. Model matriks Q dapat ditentukan dalam persamaan berikut

$$Q = \begin{bmatrix} Q_\theta & 0_{22} & 0_{22} \\ 0_{22} & Q_\phi & 0_{22} \\ 0_{22} & 0_{22} & Q_\psi \end{bmatrix} \quad (22)$$

dimana 0_{22} merupakan matriks nol 2x2 dan Q_θ , Q_ϕ , dan Q_ψ merupakan matriks observasi pada sumbu x, sumbu y, dan sumbu z. Matriks observasi sendiri dapat dituliskan dalam persamaan (23).

$$Q_i = \begin{bmatrix} \frac{dt^3}{3} & \frac{dt^2}{2} \\ \frac{dt^2}{2} & dt \end{bmatrix} \quad (23)$$

dimana dt merupakan selisih waktu dalam satuan detik.

Selanjutnya, masuk dalam proses koreksi atau *update* pengukuran *Kalman Filter*. Langkah pertama dalam proses koreksi yaitu menentukan besar nilai *kalman gain*. *Kalman gain* berfungsi untuk menentukan seberapa banyak bagian dari proses prediksi serta data dari pengukuran yang akan digunakan dan kemudian digabungkan untuk memperbarui *state* pada bagian koreksi yang dapat ditunjukkan dalam persamaan (24). (Jonathan & Rippun, 2016)

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (24)$$

dimana nilai K_k merupakan *kalman gain* dalam indeks waktu k. Nilai P_k^- prediksi kovarian *error* pada waktu k, H merupakan model matriks transisi dan R adalah *error* dari pengukuran. Model matriks transisi H dapat ditunjukkan dalam persamaan (15)

$$H = [1 \quad 0] \quad (15)$$

Apabila matriks H disimulasikan dalam program, maka akan memiliki model matriks H 3x6 dimana matriks tersebut masing-masing perkalian pada sumbu x, sumbu y, dan sumbu z sehingga model matriks H menjadi persamaan (16)

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad (16)$$

dan *error* dari pengukuran (R) dapat dirumuskan dalam persamaan (17)

$$R = \text{diag} ([\text{cov}(\dot{\theta}) \quad \text{cov}(\dot{\phi}) \quad \text{cov}(\dot{\psi})]) \quad (17)$$

Nilai R berisi diagonal dari kovarian derau sudut dari orientasi *pitch*, *roll*, *yaw* dimana $\dot{\theta}$, $\dot{\phi}$, dan $\dot{\psi}$ merupakan data kecepatan sudut pada sumbu x, sumbu y, dan sumbu z.

Langkah kedua dalam proses koreksi yaitu *update* estimasi dengan pengukuran Z_k . *State* ini sudah memuat data dari prediksi awal dan pengukuran sensor yang telah diperbarui oleh *kalman gain*. Persamaan *state* ini dapat ditunjukkan dalam persamaan (18)

$$\hat{x}_k = \hat{x}_k^- + K_k (Z_k - H \hat{x}_k^-) \quad (18)$$

dimana nilai \hat{x}_k merupakan *state* dari sistem pada waktu k, \hat{x}_k^- adalah prediksi *state* dari sistem pada waktu k, K_k adalah nilai *kalman gain*, Z_k merupakan pengukuran data dari perhitungan rumus orientasi pada persamaan 2.11, 2.12, 2.13 dan H adalah model matriks transisi. Data yang digunakan dalam perhitungan dari rumus *roll* dan *pitch* pada persamaan 2.11 dan persamaan 2.12 diambil dari sensor akselerometer dan giroskop, kemudian data untuk perhitungan yang digunakan untuk menghitung *yaw* pada persamaan 2.13 diambil dari sensor magnetometer.

Langkah yang terakhir dalam proses koreksi yaitu menghitung *update* kovarian *error* akhir yang ditunjukkan dalam persamaan (19)

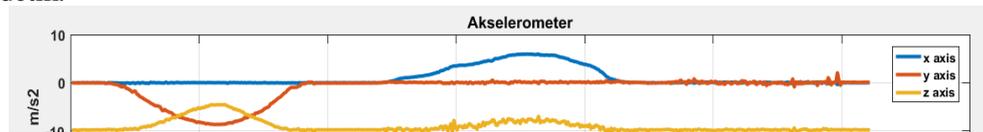
$$P_k = (I - K_k H) P_k^- \quad (19)$$

dimana P_k merupakan kovarian *error* pada waktu k, I merupakan matriks identitas 6x6, K_k adalah *kalman gain* dalam indeks waktu k, H merupakan model matriks transisi, dan P_k^- adalah prediksi kovarian *error* pada waktu k. Matriks identitas I ditunjukkan dalam persamaan 3.19

3. Hasil dan Analisa

Hasil dari percobaan ini merupakan hasil dari pergerakan sudut orientasi pitch, roll, dan yaw dengan tanpa filter dan kemudian akan di filter menggunakan kalman filter.

Percobaan dilakukan dengan pergerakan sudut tidak lebih dari 90° kearah kanan dimana menghasilkan sebanyak 557 data dari sensor akselerometer yang mengacu pada gambar 5 dengan waktu perekaman selama 31 detik.



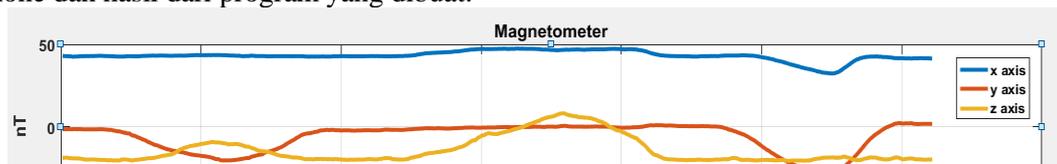
Gambar 5. Grafik Keluaran Data Sensor Akselerometer

Kemudian hasil data keluaran dari sensor giroskop yang ditunjukkan pada gambar 6 dimana masih menggunakan satuan rad/sec .



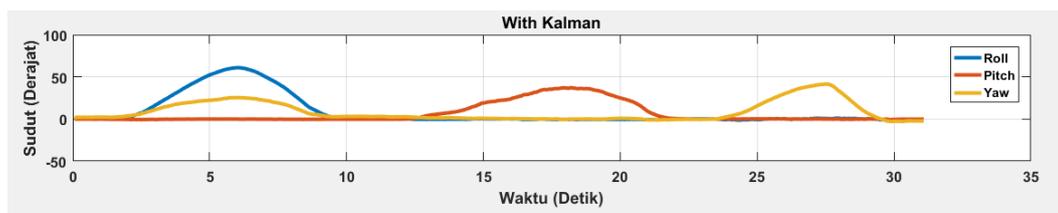
Gambar 6. Grafik Keluaran Data Sensor Giroskop

Data dari percobaan ini merupakan data yang sudah dikoversi kemudian dioalah pada matlab untuk mendapatkan hasil yang diinginkan, hasil yang didapat dapat dilihat pada gambar 7, data disajikan dalam bentuk grafik agar mempermudah pembacaan dan membandingkan antara hasil data orientasi dari smartphone dan hasil dari program yang dibuat.

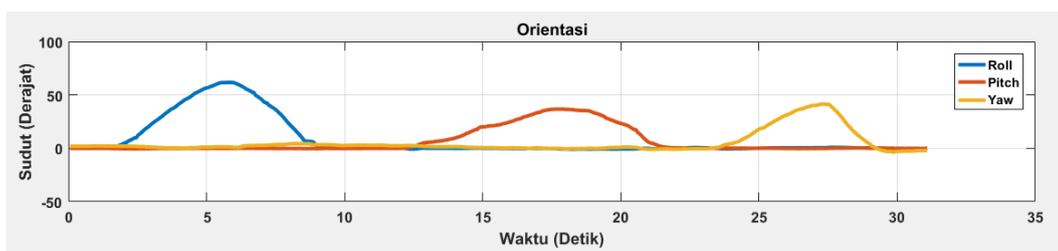


Gambar 7. Grafik Keluaran Data Sensor Magnetometer

Setelah didapat hasil keluaran dari sensor akselerometer, giroskop, dan magnetometer kemudian dilakukan proses pemfilteran dengan menggunakan *kalman filter*. Proses ini bertujuan untuk mengurangi derau atau noise pada sensor IMU serta saat perhitungannya. Data yang diproses menggunakan kalman filter tersebut kemudian diolah terlebih dahulu menggunakan matlab, dan menghasilkan data keluaran seperti pada gambar 8.



Gambar 8. Grafik Keluaran Data Gabungan Sensor Akselerometer, Sensor Giroskop, dan Sensor Magnetometer Menggunakan *Kalman Filter*



Gambar 9. Grafik Keluaran dari Orientasi pada Perangkat Lunak IMU+GPS Stream pada Smartphone

Grafik keluaran sikap roll dari perhitungan gabungan sensor akselerometer, sensor giroskop dan sensor magnetometer setelah difilter pada gambar 8 menunjukkan pergerakan sebesar $60,96^\circ$, pada grafik keluaran sikap pitch dari perhitungan gabungan sensor akselerometer, sensor giroskop dan sensor magnetometer setelah difilter menghasilkan pergerakan sudut sebesar $37,01^\circ$, grafik keluaran sikap yaw dari perhitungan gabungan sensor akselerometer, sensor giroskop dan sensor magnetometer setelah difilter menghasilkan pergerakan sudut sebesar $41,12^\circ$. Gambar 9 merupakan grafik sebagai validator dari hasil Grafik Keluaran Data Gabungan Sensor Akselerometer, Sensor Giroskop, dan Sensor Magnetometer Menggunakan *Kalman Filter*

3.1 Galat

Pada penelitian yang dilakukan ini memiliki tujuan yaitu untuk menganalisa ketepatan besar sudut orientasi dengan memanfaatkan sensor akselerometer, giroskop, dan magnetometer pada smartphone menggunakan metode Kalman Filter.

Tabel 1. Data Hasil Perhitungan Galat

	<i>Pitch</i>	<i>Roll</i>	<i>Yaw</i>
AM(%)	4.2	1.2	2.06
AGM(%)	0.5	0.8	1.4

4. Kesimpulan

Adapun kesimpulan yang didapat dalam percobaan yang sudah dilakukan, yang ditunjukkan dalam beberapa point berikut :

1. Kalman filter tidak hanya digunakan untuk mengurangi derau, tetapi juga dapat digunakan untuk menggabungkan sensor akselerometer dan magnetometer dengan sensor giroskop dikarenakan sensor giroskop tidak dapat menentukan posisi awal dalam perekaman data.
2. Pada perhitungan galat dan ketepatan rata-rata dari 3 sudut orientasi didapat hasil sebesar 0,8% galat dan ketepatan 99,2% pada sikap roll, 0,5% galat dan 99,5% ketepatan pada sikap pitch, kemudian 1,4% galat dan 98,6% ketepatan pada sikap yaw. Hasil perhitungan galat dan ketetapan sudut orientasi roll, pitch dan yaw berasal dari data keluaran gabungan sensor akselerometer dan

magnetometer dengan sensor giroskop yang sudah difilter dibandingkan dengan hasil sudut orientasi dari smartphone.

5. Daftar Pustaka

- [1] Akbar, M. A., Wahyudi, & Hidayatno, A. (2014). Perancangan Perangkat Lunak Pendeteksi Sudut dan Posisi Menggunakan Mikrokontroler Atmega 32. *Transmisi*, Vol 1, 49-53.
- [2] Alma'i, V. R., Wahyudi, & Setiawan, I. (2009). Aplikasi Sensor Accelerometer Pada Deteksi Posisi. *Makalah Seminar Tugas Akhir*, 1-10.
- [3] Ardiantara, P. S., Sumiharto, R., & Wibowo, S. B. (2014). Purwarupa Kontrol Kestabilan Posisi dan Sikap pada Pesawat Tanpa Awak Menggunakan IMU dan Algoritma Fusion Sensor Kalman Filter. *IJEIS*, Vol 4, 25-34.
- [4] Aryanto, S., Utami, E., & Sofyan, A. F. (2018). Implementasi Rigid Body pada Rigging Terhadap Animasi Dinamis Model Kendaraan Tiga Dimensi. *Jurnal Ilmiah Bidang Teknologi, ANGKASA*, Vol 10, 77-88.
- [5] Hidayatno, A., & Wahyudi. (2011). Rancang Bangun Inertial Measurement Unit Sebagai Sistem Monitoring Kendaraan Bergerak Berbasis Sensor Accelerometer dan Gyroscope. *Jurnal Rekayasa Elektrika*, Vol 9, 187-194.
- [6] Jonathan, N., & Rippun, F. (2016). Implementasi Filter Kalman Pada Sistem Sensor Inertial Measurement Unit (IMU) Quadcopter. *Jurnal Elektro*, Vol 9, 99-110.
- [7] Lasmadi. (2019). Sistem Navigasi Quadrotor Berbasis IMU dengan. *elkha*, Vol 11, 39-46.
- [8] Michel, T., Geneves, P., Fourati, H., & Layaida, N. (2017). On Attitude Estimation with Smartphones. *IEEE International Conference on Pervasive Computing and Communications (PerCom)*.
- [9] Mubarak, A., Wahyudi, & Setiawan, I. (2009). Pendeteksi Rotasi Menggunakan Gyroscope Berbasis Mikrokontroler. *Makalah Seminar Tugas Akhir*.
- [10] Noor, E. S., Rif'an, M., & Siwindarto, P. (2013). Implementasi Sensor Magnetometer dan Akselerometer Untuk Memonitor Arah Muatan Roket. *KOMURINDO*, 1-6.
- [11] Suprem, A., Deep, V., & Elarabi, T. (2017). Orientation and Displacement Detection for Smartphone Device Based IMUs. *IEEE Access*, Vol 5, 987-997.
- [12] Suryati, D. I. (2017). Inertial Measurement Unit (IMU) pada Sistem Pengendali Satelit. *Media Dirgantara*, Vol 12, 7-10.
- [13] Mudarris, Zain, S.G., (2020). Implementasi Sensor Inertial Measurement Unit (IMU) untuk Monitoring Perilaku Roket, *AVITEC*, Vol. 2, No. 1, 55-64.
- [14] Wicaksono, M. A., Kurniawan, F., & Lasmadi. (2020). Kalman Filter untuk Mengurangi Derau Sensor Accelerometer pada IMU Guna Estimasi Jarak. *AVITEC*, Vol 2, No. 2, 145-159.
- [15] Tuck, K. (2007). Tilt Sensing Using Linear Accelerometers. *Freescale Semiconductor*, Vol 2, 1-7.
- [16] Wahdan, A., Georgy, J., Abdelfatah, W. F., & Noureldin, A. (2014). Magnetometer Calibration for Portable Navigation Devices in Vehicles Using a Fast and Autonomous Technique. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, Vol 15, 2347-2352.
- [17] Matthew, R. B., Roger, S. A., & Holappa, K. (2017). A Kalman Filtering Tutorial For Undergraduate Students. *International Journal of Computer Science & Engineering Survey (IJCSSES)*, Vol 8, 1-18.
- [18] Yuliani, S., & Saputra, H. M. (2018). Kolaborasi Kalman Filter dengan Complementary Filter untuk Mengoptimasi Hasil Sensor Gyroscope dan Accelerometer. *Institut Teknoloni Nasional*, 63-68.
- [19] Gunnarsdóttir, E. L. (2012). The Earth's Magnetic Field. 10 ECTS unit research project, part of a Baccalaureus Scientiarum degree in Physics, 1-43.

- [20] Becker, A. (2018). Kalman Filter. Dipetik 1 Agustus, 2020, dari kalmanfilter.net: <https://www.kalmanfilter.net/default.aspx>
- [21] Rhudy, M. B., Salguero, R. A., & Holappa, K. (2017, Februari). A Kalman Filtering Tutorial For Undergraduate Students. IJCSES Vol. 8, No. 1, hal. 1-18.